

TUGAS AKHIR - TE 145561

**Penerapan *Field Programmable Gates Array* pada
Rangkaian *Interface Mini Computer Numerical
Control***

Bagus Kurniawan
NRP 10311500000051

Pembimbing
Slamet Budi Prayitno S.T., M.T.
Moh. Abdul Hady S.T., M.T.

DEPARTEMEN TEKNIK ELEKTRO OTOMASI
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018



TUGAS AKHIR - TE 145561

**Penerapan *Field Programmable Gates Array* pada
Rangkaian *Interface Mini Computer Numerical
Control***

Bagus Kurniawan
NRP 10311500000051

Pembimbing
Slamet Budi Prayitno S.T., M.T.
Moh. Abdul Hady S.T., M.T.

DEPARTEMEN TEKNIK ELEKTRO OTOMASI
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember
Surabaya 201



FINAL PROJECT - TE 145561

***APPLICATION OF FIELD PROGRAMMABLE GATES
ARRAY IN INTERFACE CIRCUIT MINI COMPUTER
NUMERICAL CONTROL***

Bagus Kurniawan
NRP 10311500000051

Supervisors
Slamet Budi Prayitno S.T., M.T.
Moh. Abdul Hady S.T., M.T.

DEPARTMENT OF *ELECTRICAL* AUTOMATION ENGINEERING
Faculty of Vocations
Institut Teknologi Sepuluh Nopember
Surabaya 2018

PERNYATAAN KEASLIAN PENELITIAN

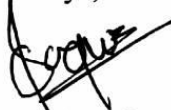
Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan dari penelitian saya dengan judul :

“Penerapan *Field Programmable Gates Array* pada Rangkaian Interface Mini Computer Numerical Control”

Adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diizinkan dan bukan merupakan karya pihak lain yang saya akui karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap didalam daftar pustaka. Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai dengan peraturan yang berlaku.

Surabaya, Juli 2018



Bagus Kurniawan

NRP. 10311500000051

---- Halaman ini sengaja dikosongkan ----

PENERAPAN *FIELD PROGRAMMABLE GATES* ARRAY PADA RANGKAIAN INTERFACE MINI COMPUTER NUMERIC CONTROL

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Memperoleh Gelar Ahli Madya Teknik
Pada
Departemen Teknik Elektro Otomasi
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember**

Pembimbing I

Menyetujui,

Pembimbing II

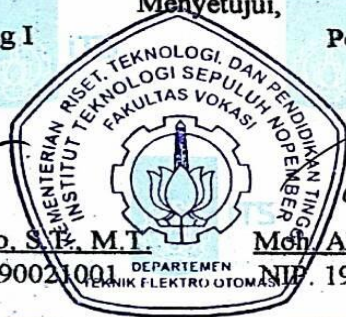
Slamet Budi Prayitno, S.T., M.T.

Moh. Abdul Hady, S.T., M.T.

NIP. 196510141990021001

**DEPARTEMEN
TEKNIK ELEKTRO OTOMASI**

NIP. 1987041320150410002



**SURABAYA
JULI, 2018**

---- Halaman ini sengaja dikosongkan ----

PENERAPAN *FIELD PROGRAMMABLE GATES ARRAY* PADA RANGKAIAN *INTERFACE* MINI *COMPUTER NUMERICAL CONTROL*

Nama : Bagus Kurniawan
Pembimbing : Slamet Budi Prayitno, S.T., M.T.
Moh. Abdul Hady, S.T., M.T.

ABSTRAK

Computer Numerical Control (CNC) merupakan sebuah pembaruan dari mesin perkakas konvensional sebelumnya yang kurang efektif dari segi waktu dan biaya. CNC memiliki komponen penyusun yaitu *servo drive*, *Programmable Logic Controller* (PLC), dan *Human Machine Interface* (HMI). CNC yang di produksi oleh PT. CNC Controller Indonesia memiliki 2 *controller* dalam mengoperasikannya, yaitu melalui HMI dan *Personal Computer* (PC). Dalam menjalankan CNC maka perlu adanya pemilihan *controller* yang digunakan, komponen ini disebut sebagai rangkaian *interface*. Rangkaian *interface* yang digunakan pada PT. CNC Controller Indonesia terbuat dari rangkaian *analog*, sehingga memiliki sifat yang tidak dapat dirubah (tidak *flexibel*).

Field Programmable Gates Array (FPGA) dapat diprogram dengan menggunakan beberapa bahasa pemrograman salah satunya yaitu menggunakan *diagram block*. FPGA dapat digunakan sebagai pengganti rangkaian *interface* yang telah di terapkan sebelumnya. FPGA dapat menggantikan beberapa komponen yang digunakan pada rangkaian *interface* seperti gerbang logika dan *buffer*.

Hasil yang didapatkan setelah menggunakan FPGA, rangkaian *interface* yang diterapkan dapat diubah sesuai yang diinginkan. Sehingga, jika terjadi pergantian sistem yang digunakan atau terjadi sebuah kesalahan maka dapat diatasi dengan cara mengubah program yang tertanam di dalam FPGA.

Kata Kunci : *Computer Numerically Control* (CNC), Rangkaian *Interface*, *Field Programmable Gates Array* (FPGA)

-----Halaman ini sengaja dikosongkan---

APPLICATION OF FIELD PROGRAMMABLE GATES ARRAY IN INTERFACE CIRCUIT MINI COMPUTER NUMERICAL CONTROL

Name : Bagus Kurniawan
Supervisor : Slamet Budi Prayitno, S.T., M.T.
Moh. Abdul Hady, S.T., M.T.

ABSTRACT

Computer Numerical Control (CNC) is an invention of previous conventional machine tools that are less cost-effective in terms of time and cost. CNC has a constituent component that is servo drive, Programmable Logic Controller (PLC), and Human Ma-chine Interface (HMI). CNC produced by PT. CNC Controller Indonesia has 2 controllers in operation, ie through HMI and Personal Computer (PC). In running the CNC then it is necessary to select the controller used, this component is called interface circuit. The series of interface used in PT. CNC Controller Indonesia is made of analog circuits, so it has irreversible properties (not flexsibel).

Field Programmable Gates Array (FPGA) can be programmed using several programming languages, one of them is using diagram block. FPGA can be used as a substitute for interface that have been applied before. FPGA can replace some of the components used in interface such as logic gates and buffers.

Expected results after using the FPGA, the interface circuit applied can be changed as desired. Se-up, if there is a change of sistem used or happened as a mistake then can be overcome by change the program embedded inside FPGA.

Keywords: Computer Numerically Control (CNC), Interface Circuit, Field Programmable Gates Array (FPGA)

-----Halaman ini sengaja dikosongkan-----

KATA PENGANTAR

Puji syukur atas kehadiran Tuhan Yang Maha Esa karena atas rahmat dan karunianya penulis dapat menyelesaikan Penelitian dengan judul **“Penerapan *Field Programmable Gates Array* pada Rangkaian *Interface Mini Computer Numerical Control*”** untuk memenuhi sebagian persyaratan guna menyelesaikan pendidikan Diploma 3 di Departemen Teknik Elektro Otomasi, Fakultas Vokasi, Institut Teknologi Sepuluh Nopember Surabaya

Dalam proses penyusunannya penulis telah banyak dibantu oleh berbagai pihak. Oleh karena itu pada kesempatan ini penulis mengucapkan terimakasih kepada pihak keluarga yang selalu memberi dukungan dan doa untuk keberhasilan penulis. Bapak Slamet Budi Prayitno, S.T., M.T dan Bapak Moh. Abdul Hady, S.T., M.T., selaku Dosen Pembimbing Penelitian di Institut Teknologi Sepuluh Nopember. Lalu kepada Bapak Imam Syaefudin selaku pembimbing penelitian di PT CNC Controller Indonesia dan semua pihak yang telah membantu secara langsung maupun tidak langsung dalam proses penyelesaian penelitian ini. Keluarga laboratorium Sistem Komputer dan Otomasi angkatan 2015, 2016 dan 2017 atas kerja samanya. Teman-teman dan semua pihak yang tidak dapat penulis sebutkan satu persatu.

Penulis menyadari bahwa buku penelitian akhir ini belum sempurna, oleh karena itu saran dan masukan sangat diharapkan untuk perbaikan di masa yang akan datang. Semoga buku ini bermanfaat bagi pembaca dan masyarakat pada umumnya.

Surabaya, 25 Juli 2018

Bagus Kurniawan
NRP 10311500000051

-----Halaman ini sengaja dikosongkan-----

DAFTAR ISI

HALAMAN PENGESAHAN	ix
ABSTRAK.....	xi
KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR.....	xix
DAFTAR TABEL	xxi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Tujuan.....	2
1.3 Rumusan Masalah	2
1.4 Batasan Masalah.....	2
1.5 Metodologi	2
1.6 Sistematika Penulisan	3
BAB II TEORI PENUNJANG CNC, RANGKAIAN <i>INTERFACE</i> , DAN FPGA	5
2.1 Computer Numerical Control	5
2.2 Teori <i>Field Programmable Gates Array</i> (FPGA).....	20
2.3 Quartus Prime	24
2.4 <i>Software Eagle</i>	26
BAB III PERANCANGAN RANGKAIAN <i>INTERFACE</i> DAN PEMOGRAMAN FPGA	29
3.1 Pemetaan <i>Input</i> dan <i>Output</i>	31
3.2 Pemetaan Tegangan <i>Input</i> dan <i>Output</i>	36
3.3 Pembuatan Schematic Rangkaian.....	38
3.4 Pemograman <i>Field Programmable Gates Array</i> (FPGA)	39
3.5 Pemasangan Rangkaian ke Mesin CNC MTU.....	39
BAB IV PENGUJIAN DAN ANALISIS	43
4.1 Pengujian <i>Field Programmable Gates Array</i> (FPGA)	43
4.2 Pengujian Rangkaian <i>Optocoupler</i>	46
4.3 Pengujian Rangkaian <i>Relay</i>	47
4.4 Pengujian Rangkaian <i>Interface</i> pada mesin mini CNC	47
BAB V PENUTUP	51
DAFTAR PUSTAKA	53
LAMPIRAN A.....	55
LAMPIRAN B.....	57
LAMPIRAN C.....	61
BIODATA PENULIS	63

--- Halaman ini sengaja dikosongkan ---

DAFTAR GAMBAR

Gambar 2.1	HMI Sinumerik 808D.....	8
Gambar 2.2	Tampak Belakang SIEMENS PPU 808D.....	8
Gambar 2.3	Servo Drive ASDA A2.....	10
Gambar 2.4	Rangkaian Dalam Servo Drive ASDA A2 [1].....	12
Gambar 2.5	Koneksi pada <i>Tower lamp</i>	13
Gambar 2.6	Konstruksi <i>Relay Normally open</i>	16
Gambar 2.7	Konstruksi <i>Relay Normally Close</i>	16
Gambar 2.8	<i>Optocoupler</i>	19
Gambar 2.9	Struktur Dalam FPGA [3].....	22
Gambar 2.10	Rangkaian Dalam <i>Logic cell</i>	23
Gambar 2.11	Bentuk Fisik FPGA CYCLONE IV E.....	25
Gambar 2.12	Tampilan Awal Quartus Prime.....	26
Gambar 2.13	Tampilan Jendela Pemograman <i>Diagram block</i>	26
Gambar 2.14	Tampilan Awal EAGLE.....	27
Gambar 2.15	Lembar Kerja Desain <i>Schematic</i>	27
Gambar 2.16	Lembar Kerja Desain PCB.....	28
Gambar 3.1	Flowchart Perancangan dan Implementasi Alat.....	30
Gambar 4.1	Program Pada <i>Tower lamp</i>	44
Gambar 4.2	Rangkaian <i>Optocoupler</i>	47
Gambar 4.3	Rangkaian <i>Relay</i>	47

--- Halaman ini sengaja dikosongkan ---

DAFTAR TABEL

Tabel 2.1 Keterangan Pin <i>Port</i> DB25	6
Tabel 2.2 Keterangan <i>Port</i> pada SINUMERIK PPU 808D.....	9
Tabel 2.3 Karakteristik FPGA CYCLONE IVE	25
Tabel 3.1 Pemetaan <i>Input Output</i> Rangkaian <i>Interface</i>	31
Tabel 3.2 Pin yang terhubung pada rangkaian <i>interface</i> pada HMI...	33
Tabel 3.3 Penjabaran I/O <i>Programmable Logic Controller</i>	33
Tabel 3.4 Penjabaran Pin <i>Port</i> DB 25 <i>Computer</i>	34
Tabel 3.5 Penjabaran Pin <i>Servo Drive</i>	36
Tabel 3.6 Pemetaan Tegangan I/O yang dibutuhkan	36
Tabel 3.7 Pemetaan I/O Rangkaian <i>Interface</i>	39
Tabel 4.1 Daftar <i>Input</i> dan <i>Output</i> pemograman FPGA	44
Tabel 4.2 Data percobaan rangkaian <i>tower lamp</i>	45
Tabel 4.3 Tabel <i>Input</i> dan <i>Output</i> Rangkaian <i>Pulse</i>	46
Tabel 4.4 Konfigurasi Pin FPGA pada Rangkaian <i>Interface</i>	48

-----Halaman ini sengaja dikosongkan---

BAB I

PENDAHULUAN

1.1 Latar Belakang

PT. CNC *Controller* merupakan perusahaan yang bergerak dalam bidang manufaktur yang memproduksi mesin mini *Computer Numerical Control* (CNC). Perusahaan ini memiliki beberapa divisi diantaranya adalah divisi *electric* dan divisi *mechanical*. Divisi *electrical* memiliki tugas dalam mendesain *electric* pada mesin mini CNC sehingga mesin tersebut dapat berfungsi dengan baik. Selain mendesain bagian *electrical* pada mesin mini CNC, divisi *electrical* memiliki tugas seperti *wiring* elemen – elemen penyusun dari mesin mini CNC, pemasangan komponen penyusun CNC kedalam mesin mini CNC seperti pemasangan *servo drive*, *Human Machine Interface* (HMI), dan pemasangan komponen lainnya yang akan dihubungkan, sehingga mesin mini CNC dapat berjalan dengan baik.

Mesin mini CNC memiliki perbedaan dengan mesin CNC pada umumnya. Mesin mini CNC memiliki 2 buah *controller* yaitu menggunakan HMI dengan tipe Sinumerik 808D, dan menggunakan *Personal Computer* (PC) dalam menjalankan mesin tersebut. Mesin mini CNC tidak dapat bergerak dengan menggunakan kedua *controller* tersebut secara bersamaan. Oleh karena itu diperlukan sebuah elemen yang berfungsi agar mesin tersebut tidak dapat berjalan dengan menggunakan 2 *controller* sekaligus. Elemen yang dimaksud adalah rangkaian *interface*. Rangkaian *interface* tersusun dari beberapa komponen yaitu *optocoupler*, *relay*, dan gerbang-gerbang logika. Rangkaian *interface* yang dibuat oleh divisi *electrical* PT. CNC Controller Indonesia masih bersifat *analog*. Sifat yang tidak dapat berubah-ubah jika terjadi sebuah perubahan pada rangkaian merupakan salah satu kekurangan dari rangkaian *interface*.

Field Programmable Gates Array (FPGA) merupakan sebuah *embedded controller* yang dapat diprogram dengan menggunakan beberapa bahasa pemrograman salah satunya yaitu menggunakan *diagram block*. FPGA terdiri dari *Integrated Circuit* (IC) yang berisikan gerbang-gerbang logika dan elemen rangkaian digital lainnya seperti *comparator*, *buffer*, dan elemen lainnya. FPGA dapat digunakan sebagai pengganti rangkaian *interface* yang telah di terapkan sebelumnya. FPGA dapat menggantikan beberapa komponen yang digunakan pada rangkaian *interface* seperti komparator, gerbang logika dan *buffer*.

Hasil yang didapatkan setelah menggunakan FPGA, rangkaian *interface* yang diterapkan dapat dirubah sesuai kondisi yang diinginkan oleh perusahaan. Sehingga, jika terjadi pergantian *sistem* yang digunakan atau terjadi sebuah kesalahan maka dapat diatasi dengan cara merubah program yang tertanam di dalam FPGA.

1.2 Tujuan

Tujuan dari penelitian ini adalah penerapan FPGA sebagai pengganti *Integrated Circuit* (IC) gerbang logika yang digunakan pada rangkaian sebelumnya. Serta membuat rangkaian *interface* yang dapat diganti apabila terjadi sebuah kesalahan atau perubahan dalam perancangan *sistem*.

1.3 Rumusan Masalah

Rangkaian *interface* yang terdiri dari beberapa *Integrated Circuit* (IC) dan komponen analog lainnya yang memiliki sifat tetap pada rangkaian yang telah dibuat. Perubahan rangkaian akan terjadi apabila terjadi perubahan elemen penyusun yang digunakan atau adanya sebuah kesalahan dalam perancangan mesin mini CNC. Sehingga rangkaian *interface* yang dibuat saat ini tidak fleksibel.

1.4 Batasan Masalah

Penelitian ini menggunakan bahasa pemrograman diagram *block* pada FPGA. Rangkaian *interface* yang dibuat hanya dapat menampung *input* dan *output* sebesar 5V dan 24V.

1.5 Metodologi

Penelitian ini dilakukan melalui beberapa tahapan metodologi, yaitu, tahap persiapan, tahap perencanaan dan pembuatan alat, tahap pengujian dan analisis, dan yang terakhir adalah penyusunan laporan berupa buku Penelitian. Pada tahap persiapan telah dilakukannya studi literatur terhadap rangkaian *interface* yang digunakan di PT.CNC Controller Indonesia. Hal ini meliputi koneksi *input* dan *output* yang terhubung dengan rangkaian *interface*, dan besarnya tegangan yang masuk dan yang akan diberikan dan diterima oleh masing masing *input output* (I/O). Selain mempelajari koneksi dari rangkaian *interface*, diperlukan pula mempelajari mengenai FPGA yang akan digunakan baik itu berupa tegangan yang dapat digunakan sebagai I/O dan mempertimbangkan jumlah I/O yang digunakan di FPGA yang akan

diterapkan di rangkaian *interface*. Pada tahap perencanaan dan pembuatan alat, telah dilakukan desain rangkaian rangkaian *interface* yang akan digunakan menggunakan FPGA. Tahap ini merupakan tahap penentuan *input* dan *output* yang akan dimasukkan dengan FPGA, dan penentuan apakah *input* dan *output* yang digunakan membutuhkan sebuah *converter* tegangan. Tahap pengujian dan analisa alat, pemasangan rangkaian *interface* yang sudah didesain ke mesin mini CNC. Tahap ini merupakan tahap pengujian alat yang telah dibuat, apakah alat tersebut dapat berjalan dengan baik atau tidak. Tahap terakhir yaitu tahap penyusunan laporan, penulisan laporan berupa buku Penelitian. Data data yang ditulis merupakan data yang didapatkan dari tahap pengujian dan analisa.

1.6 Sistematika Penulisan

Pembahasan Penelitian ini akan dibagi menjadi lima Bab dengan sistematika sebagai berikut:

BAB I PENDAHULUAN

Bab ini meliputi latar belakang, permasalahan, tujuan penelitian, metodologi penelitian, dan sistematika laporan Penelitian yang akan dibuat.

BAB II TEORI PENUNJANG CNC, RANGKAIAN INTERFACE, DAN FPGA.

Bab ini menjelaskan mengenai teori penunjang alat Penelitian yang akan dibuat meliputi teori mengenai CNC, rangkaian *interface*, elemen penyusun rangkaian *interface* dan teori mengenai FPGA.

BAB III PERANCANGAN RANGKAIAN INTERFACE

Bab ini membahas perencanaan dan pembuatan perangkat keras (*Hardware*) yang meliputi desain alat, dan program alat yang digunakan pada Penelitian ini.

BAB IV PENGUJIAN DAN ANALISIS ALAT

Menjelaskan tata cara pengujian alat yang telah dibuat. mengambil dan menganalisis data rangkaian *Interface* saat dipasang ke mesin mini CNC. Menganalisis perbandingan data rangkaian *interface* versi lama dengan versi baru

BAB V PENUTUP

Bab ini berisi kesimpulan dan saran dari hasil percobaan yang telah diperoleh.

-----Halaman ini sengaja dikosongkan---

BAB II

TEORI PENUNJANG CNC, RANGKAIAN INTERFACE, DAN FPGA

Beberapa teori yang dijelaskan dalam bab ini adalah penunjang untuk mengerjakan Penelitian. Penjelasan yang dibahas tentang teori dasar CNC, Mesin mini CNC, Rangkaian *interface*, *optocoupler*, *relay* dan FPGA. Teori-teori yang telah diambil berasal dari berbagai macam literatur.

2.1 Computer Numerical Control

Sistem otomatisasi mesin perkakas yang dioperasikan oleh perintah yang diprogram dalam bentuk kode angka dan disimpan dimedia penyimpanan disebut *Computer Numerical Control* (CNC). Karena informasi yang digunakan berbentuk rumus matematik, maka sistem ini dinamakan *control* dengan angka (*numerical control*).

Mesin CNC pertama kali digunakan dalam proses mengikir, membuat lubang, memutar, mengasah dan menggergaji, dan tahun – tahun terakhir ini juga digunakan untuk membengkokkan pipa dan membuat berbagai bentuk. Dibandingkan dengan peralatan biasa, mesin yang dikontrol dengan kode angka ini lebih cermat, cepat, konsisten dan fleksibel, bahkan untuk *manufacturing* yang sangat rumit sekalipun. Rancangan produk dapat diubah atau disesuaikan cukup dengan mengubah instruksi.

Awal lahirnya mesin CNC bermula dari 1952 yang dikembangkan oleh John Pearseon dari Institut Teknologi Massachusetts, atas nama Angkatan Udara Amerika Serikat. Semula proyek tersebut diperuntukkan untuk membuat benda kerja khusus yang rumit. Semula perangkat mesin CNC memerlukan biaya yang tinggi dan volume unit pengendali yang besar.

Pada tahun 1973, mesin CNC masih sangat mahal sehingga masih sedikit perusahaan yang mempunyai keberanian dalam memelopori *investasi* dalam teknologi ini. Dari tahun 1975, produksi mesin CNC mulai berkembang pesat. Perkembangan ini dipacu oleh perkembangan *microprosesor*, sehingga volume unit pengendali dapat lebih ringkas.

Dewasa ini penggunaan mesin CNC hampir terdapat di segala bidang. Dari bidang pendidikan dan riset yang mempergunakan alat-alat demikian dihasilkan berbagai hasil penelitian yang bermanfaat yang sudah banyak digunakan dalam kehidupan sehari-hari masyarakat banyak. [1]

Mesin mini *CNC MTU* ini adalah salah satu jenis dari mesin CNC *training unit* yang diproduksi oleh PT.CNC Controller Indoneisa. Mesin CNC *training unit* yang diproduksi oleh perusahaan ini memiliki 2 *controller* yaitu dengan menggunakan *Personal Computer* (PC) dan *Human Machine Interface* (HMI). Mesin mini CNC juga memiliki beberapa komponen penyusun lainnya yaitu *Programmable Logic Controller*, *Servo Drive* dan *Tower lamp*.

Personal Computer (PC) digunakan sebagai *controller* pada mesin mini CNC. Dalam *control* mesin CNC dengan menggunakan mode PC menggunakan sebuah perangkat lunak (*software*) yang bernama Mach3. Komunikasi PC dengan elemen lainnya menggunakan *Port* parallel DB 25. Adapun penjabaran dari setiap pin pada DB25 dapat dilihat pada Tabel 2.1. Pin pada DB 25 *Port parallel* PC memiliki fungsi masing masing untuk memberikan sebuah informasi dari *control* PC ke elemen lainnya yaitu *servo drive*, PLC, dan *tower lamp*. Servo On merupakan sebuah status yang diberikan oleh PC untuk memerintahkan *driver* agar menghidupkan servo pada setiap *axis* yang ada. Untuk menggerakkan servo pada setiap *axis* maka diperlukannya *pulse*. *Pulse* DIR, dan STEP merupakan sebuah *pulse* yang diberikan oleh PC untuk menggerakkan servo motor. *Port* parallel DB25 ini juga akan memberikan sinyal informasi untuk memberikan sinyal menghidupkan *worklamp* atau *tower lamp* yang akan digunakan. Indikator pada *towerlamp* ada 3 jenis yaitu merah merupakan kondisi mesin dalam keadaan *error*, kuning merupakan kondisi dimana mesin dalam kondisi *standby*, dan hijau merupakan kondisi dimana mesin dalam kondisi *run*. Pada *port parallel* juga akan memberikan informasi apabila tombol *emergency* pada *software* mach3 ditekan maka akan memberikan sebuah sinyal untuk memberhentikan pergerakan dari motor servo dan memberikan indikator pada *tower lamp*.

Tabel 2.1 Keterangan Pin *Port* DB25 [6]

PIN KE -	SYMBOL	VOLTAGE
1	Servo ON	+5V
2	<i>Pulse</i> DIR Axis Y	+5V

Tabel 2.1 Keterangan Pin Port DB25 (Lanjutan)

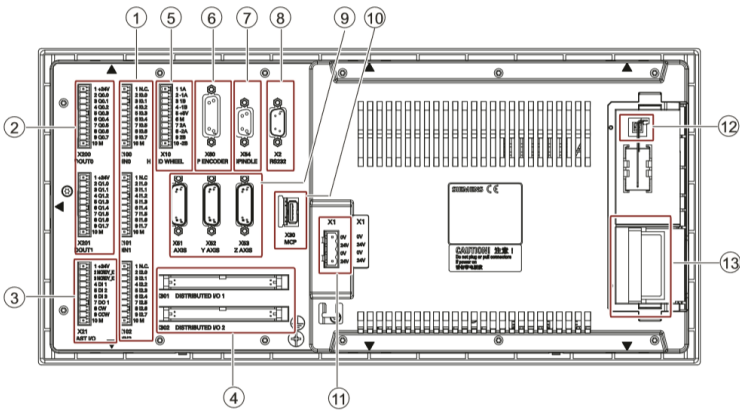
PIN KE-	SYMBOL	VOLTAGE
3	<i>Pulse STEP Axis Y</i>	+5V
4	<i>Pulse DIR Axis X</i>	+5V
5	<i>Pulse STEP Axis X</i>	+5V
6	<i>Work Lamp 1</i>	+5V
7	<i>Work Lamp 2</i>	+5V
8	<i>Work Lamp 3</i>	+5V
9	-	-
10	<i>Emergency</i>	+5V
11	-	-
12	-	-
13	-	-
14	<i>Spindle Run</i>	+5V
15	-	-
16	<i>Pulse STEP Axis Z</i>	+5V
17	<i>Pulse DIR Axis Z</i>	+5V
18	-	-
19	-	-
20	-	-
21	-	-
22	-	-
23	-	-
24	-	-
25	GND	GND

Selain menggunakan PC, Mini CNC juga menggunakan *Human Machine Interface* (HMI), yang merupakan sebuah perantara antara operator dan mesin agar dapat saling berinteraksi. Pada panel *control* terdapat beberapa perintah khusus yang dapat mengontrol mesin, seperti menggerakkan *spindle*, menggerakkan meja objek, dan mengatur kecepatan pada motor dengan mengubah *setting* parameter. Pada mesin mini CNC yang diproduksi oleh PT.CNC Controller menggunakan jenis HMI Siemens dengan tipe SINUMERIK 808D. Berikut gambar dari SINUMERIK 808D tampak depan dapat dilihat pada Gambar 2.1. Tampak belakang dari sinumerik merupakan bagian *port* yang akan terhubung dengan elemen-elemen lainnya. Gambar 2.2 merupakan

tampak SINUMERIK 808D dari belakang, dimana penjelasan dari setiap *port* yang digunakan akan dijelaskan pada Tabel 2.2.



Gambar 2.1 HMI Sinumerik 808D



Gambar 2.2 Tampak Belakang SIEMENS PPU 808D

Pada PPU 808D terdapat beberapa *port* yang akan disambung menuju elemen lain, dimana *port*-*port* yang ada memiliki fungsi yang berbeda-beda. *Port* pada *pulse drive interface* memiliki tegangan sebesar 5V, sedangkan pada digital *input-output* (I/O) memiliki tegangan sebesar 24V.

Tabel 2.2 Keterangan *port* pada SINUMERIK PPU 808D [2]

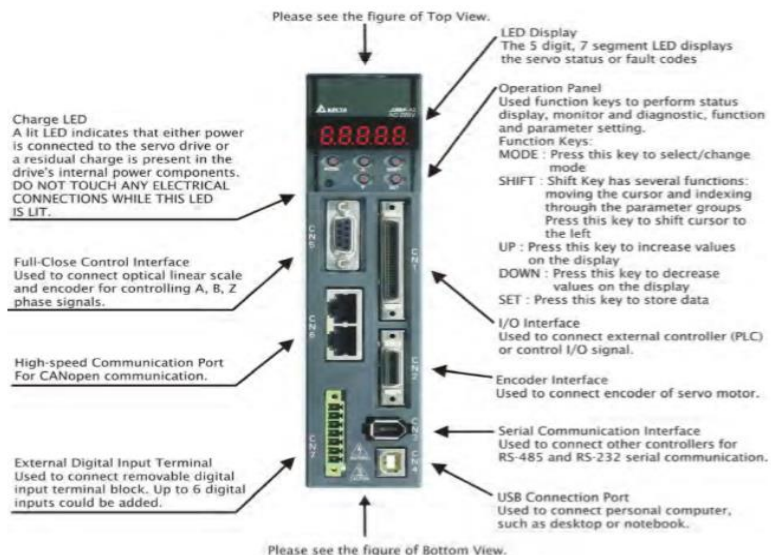
Label	Interface	Keterangan
1	X100, X101, X102	Digital Inputs
2	X200, X201	Digital Outputs
3	X21	Fast I/O
4	X301, X302	Distributed I/O
5	X10	Hand-Wheel Input
6	X60	Spindle Encoder interface
7	X54	Analog Spindle interface
8	X2	RS232 interface
9	X51, X52, X53	Pulse Drive Interface
10	X30	USB Interface, for connection with MCP
11	X1	Power Supply (+24 V)
12	-	Battery
13	-	Slot for the Sistem CompactFlash Card (CF Card)

Salah satu elemen penyusun mini CNC adalah servo *drive*, merupakan komponen yang paling terpenting pada CNC. Servo *drive* memiliki fungsi sebagai penggerak dari motor servo dan memberikan informasi baik berupa sebuah status maupun sebuah *alarm* yang sedang dialami oleh servo motor. Status dan *alarm* yang dikirim oleh servo *drive* akan diteruskan menuju HMI atau menuju PC. Servo *drive* yang digunakan pada PT. CNC Controller Indonesia terdapat 3 jenis *driver* yaitu siemens, delta dan sanyo denki. Tampak dari servo *drive* ASDA A2 dapat dilihat pada Gambar 2.3, dan rangkaian dalam dari servo *drive* ASDA A2 dapat dilihat pada Gambar 2.4.

Signal general dari servo *drive* ASDA A2 adalah sebagai berikut [2] :

- *Analog Signal Input*, Sebagai pengaturan kecepatan dengan tegangan -10V hingga +10V.
- *Analog Monitor Output*, Memberikan tegangan analog ke servo motor untuk mengatur tegangan dan arus.
- *Position Pulse Input (PULSE, /PULSE, SIGN, /SIGN)*. Servo *drive* akan menerima 2 *pulse input* , yaitu dari *Line-driver* dan *open collector*.
- *High-speed Position Pulse Input (HSIGN, /HSIGN, HPULSE, /HPULSE)*. Servo *drive* dapat menerima 2 tipe dari *High-speed pulse inputs*: +5v *input* dan *Line-driver input*. *Input* frekuensi maksimum adalah 4 MHz.

- *Position Pulse Output* (OA, /OA, OB, /OB, OZ, /OZ) merupakan keluaran dari *signal encoder* dan (OCZ) merupakan *signal output* dari *open-collector output*.
- *Power* (VDD, COM+, COM-, VCC, GND). VDD memiliki nilai maksimum +24V dengan arus 500mA. COM+ (Tegangan positif pada *digital input* dan *digital output* sebesar VDD) dan COM- (Tegangan *negative* pada COM+). VCC sebesar 12V untuk *analog*.



Gambar 2.3 Servo Drive ASDA A2

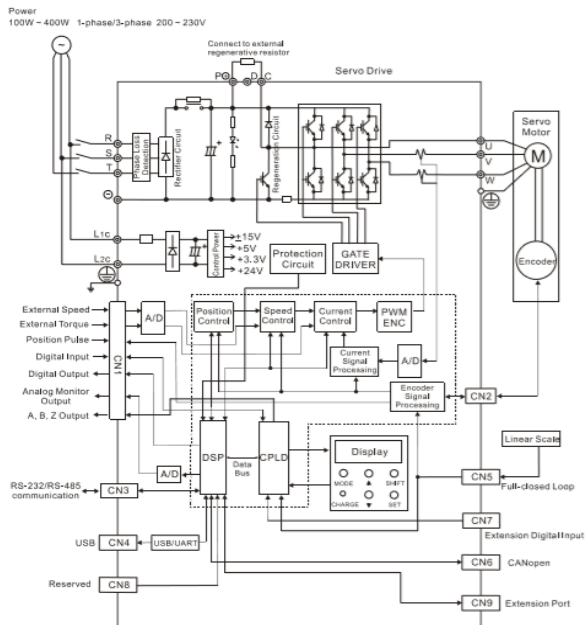
Digital output signal yang dihasilkan dari *servo drive* adalah sebagai berikut :

- **SRDY**(Servo Ready) merupakan signal DO yang akan aktif saat servo drive siap untuk *run*.
- **SON**(Servo On) merupakan signal DO yang akan aktif saat *control power* sudah terhubung dengan *servo drive*.
- **ZSPD**(Zero Speed) merupakan signal DO yang akan aktif saat servo drive mendapatkan informasi bahwa kecepatan motor dibawah 10 rpm.

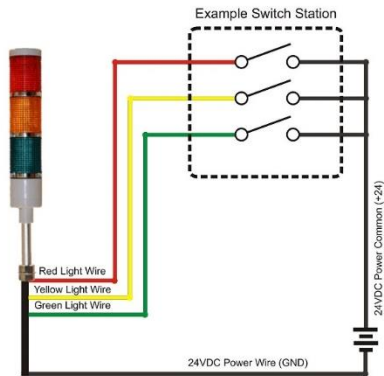
- TSPD merupakan signal DO yang akan aktif saat kecepatan motor dibawah *target rotation speed*.
- TPOS merupakan signal DO yang akan aktif saat posisi *error* sama atau diantara dengan yang diatur. Contoh , TPOS akan aktif saat motor diantara -99 dan 99 *pulse*.
- TQL merupakan signal DO yang akan aktif saat *torque* melebihi batas yang ditentukan.
- ALRM merupakan signal DO yang akan aktif saat kondisi yang gagal/salah (*serial komunikasi error, emergency stop*, dan lain lain).
- BRKR merupakan signal DO yang akan aktif saat motor melakukan kondisi rem.
- HOME merupakan signal DO yang akan aktif saat sensor *home* (sensor *limit* yang berada pada koordinat 0 pada setiap koordinat) telah aktif.
- OLW merupakan signal DO yang akan aktif saat motor mencapai tingkat *overload output*.
- WARN merupakan signal DO yang akan aktif saat *servo drive* mendeteksi adanya *ALARM* pada *servo drive* seperti *emergency stop, serial communication error*, dan lainnya.
- OVF merupakan signal DO yang akan aktif saat posisi yang dikeluarkan oleh motor melebihi batas.
- SNL (SCWL) dan SPL (SCCWL) merupakan signal DO yang akan aktif saat *servo drive* mendeteksi *reverse/forward* batasan tercapai.
- CMD_OK merupakan signal DO yang akan aktif saat perintah posisi motor selesai.
- CAP_OK merupakan signal DO yang akan aktif saat *servo drive* mendeteksi adanya perintah yang telah terlaksana.
- MC_OK (*Motion Control Completed*) merupakan signal DO yang akan aktif saat perintah posisi telah diberikan dn penentuan posisi telah selesai.
- CAM_AREA merupakan signal DO diaktifkan saat *servo drive* mendeteksi posisi *master E-CAM* (CAM elektronik) berada dalam area pengaturan.
- SP OK merupakan signal DO diaktifkan saat *error* kecepatan sama dan diantara pengaturan yang telah diberikan.
- SDO_x merupakan keluaran dari bit xx. Dimana x adalah bilangan *hexadecimal C* dari 0 – F.

Digital input signal yang diperlukan oleh *servo drive* adalah sebagai berikut :

- SON (*Servo On*) untuk memberikan *digital input* dan mengaktifkan SRDY.
- ARST digunakan untuk menghapus nomor *alarm*.
- CCLR digunakan untuk menghapus *pulse*.
- ZCLAMP digunakan untuk mengunci motor apabila nilai kecepatan kurang dari nilai yang diberikan.
- CMDINV digunakan untuk memutar balik putaran motor.
- CTRG digunakan untuk men-trigger *rising edge of the pulse*.
- TRQLM, aktif saat batas torsi benar.
- SPDLM, aktif saat batas kecepatan benar.
- STOP, aktif apabila kondisi motor *stop*.



Gambar 2.4 Rangkaian Dalam Servo Drive ASDA A2 [1]



Gambar 2.5 Koneksi pada *Tower lamp*

Tower lamp digunakan sebagai *indicator* pada CNC. *Tower lamp* terdiri dari 3 jenis warna LED yaitu hijau yang menandakan bahwa mesin dalam kondisi bekerja, kuning yang menandakan bahwa mesin dalam kondisi *ready*, dan merah yang menandakan bahwa mesin dalam kondisi *error*. Wire pada *tower lamp* disambungkan ke rangkaian *interface* dan aktif dengan membandingkan kondisi pada servo drive dari PPU 808D atau *Personal Computer* (PC).

Penyusun mesin mini CNC terdapat *Programmable Logic Controller* terdiri dari beberapa kata yaitu *Programmable*, menunjukkan kemampuan untuk menyimpan *program* yang telah dibuat ke dalam *memory* dan dapat diubah-ubah fungsi atau kegunaannya. *Logic*, menunjukkan kemampuan dalam memproses *input* secara *aritmatik* dan *logic*, yakni melakukan operasi membandingkan, menjumlahkan, mengalikan, membagi, mengurangi, negasi, dan lain sebagainya. *Controller*, menunjukkan kemampuan dalam mengontrol dan mengatur proses sehingga menghasilkan *output* yang diinginkan. Sedangkan menurut National Electrical Manufacturing Assosiation (NEMA), PLC merupakan suatu perangkat elektronik digital dengan memori yang dapat diprogram untuk menyimpan instruksi-instruksi yang menjalankan fungsi-fungsi spesifik seperti : logika, *timer*, *counter*, dan aritmatika untuk mengendalikan suatu mesin industri atau proses industri sesuai dengan yang diinginkan.

Fungsi khusus PLC adalah sebagai pemberik masukan (*input*) ke CNC untuk kepentingan pemrosesan lebih lanjut. PLC digunakan sebagai memberikan sebuah informasi yang dibutuhkan oleh *Numerical*

Control (NC). Pada HMI PPU 808D terdapat *port digital input* dan *port digital output*, dimana *port* inilah yang akan disambungkan ke PLC. Selain berasal dari HMI PPU 808D, PLC juga akan menerima informasi yang berasal dari *port* parallel PC.

PLC yang digunakan pada PT CNC Controller dalam membangun sistem mesin mini CNC menggunakan PLC S7-1200 dengan *input* dan *output* sebanyak 28 (Pin *input* sebanyak 16, pin *output* sebanyak 12). Komunikasi PLC dengan *controller* menggunakan kabel LAN, dimana PLC dapat diprogram langsung melalui HMI PPU 808D atau menggunakan *personal computer* (PC).

2.1 Teori Rangkaian Interface

Elemen penyusun CNC yang berfungsi mengalihkan mode yang akan digunakan pada mesin mini CNC yang diproduksi oleh PT CNC Controller Indonesia adalah rangkaian *interface*. Rangkaian ini terdiri dari beberapa komponen yang membentuk sebuah sistem pemisahan data yang dikirim oleh *controller*. Data yang dikirim dari *controller* ke *driver* atau sebaliknya berupa sebuah status kondisi dari mesin mini CNC, *pulse* penggerak motor, dan *status* kondisi lainnya. Pada Lampiran A1 merupakan sebuah rangkaian pemisah data PC atau HMI Siemens Sinumerik 808D menuju *tower lamp*.

Pemisah data kondisi mesin mini CNC yang akan dikirim ke *tower lamp* dari 1 *controller* baik itu dari HMI Siemens sinumerik 808D atau melalui PC mengalami sebuah pembandingan data. Terdapat 3 *comparator* untuk membandingkan data dari setiap kondisi, yaitu data *ready*, *run* dan *error*. Gerbang logika yang terdapat pada rangkaian berfungsi untuk memilah data, data yang berasal dari PC atau data dari HMI Sinumerik 808D. Data *ready* akan dikirimkan dari *controller* ke *tower lamp* untuk menyalakan *lampu* kuning, sedangkan data kondisi mesin *run* akan menyalakan *lampu* hijau dari *tower lamp*, dan data kondisi mesin *error* akan menyalakan *lampu* merah dari *tower lamp*. Selain berfungsi sebagai pemilah data kondisi dari mesin mini CNC, rangkaian *interface* berfungsi sebagai pemilah *pulse* yang akan masuk ke setiap *servo drive*.

Pulse yang masuk kedalam *servo drive* harus berasal dari 1 *controller*, baik itu berasal dari HMI Sinumerik 808D atau berasal dari *personal computer*. Terdapat 12 pin *input* yang masuk kedalam *servo drive* untuk menggerakkan motor servo pada mesin mini CNC. Lampiran B.3 merupakan gambar rangkaian pemisah *pulse* motor dari 2

controller yang tersedia. Selain itu, rangkaian *interface* berfungsi pula sebagai pemberi *pulse* z dari servo *drive* ke HMI Sinumerik 808D untuk memastikan posisi *home* (posisi koordinat 0) dari setiap motor servo. Selain itu, rangkaian *interface* juga berfungsi untuk memberikan berbagai informasi yang dibutuhkan oleh *controller* atau servo *drive*. Lampiran B.2 merupakan rangkaian pemberian informasi dari atau ke setiap motor.

Informasi yang didapatkan dan diterima oleh servo *drive* maupun *controller* berupa sebuah informasi keadaan dari servo motor. Informasi tersebut dapat berupa tegangan 24V, 5V bahkan 0V. Adapun informasi yang dibutuhkan adalah kondisi *alarm*, servo *on*, dan servo *ready*.

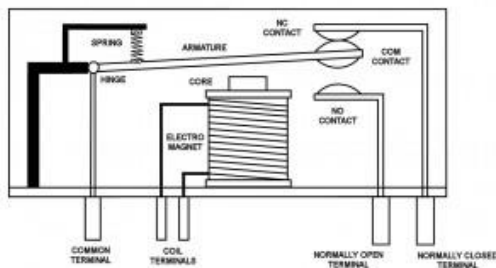
Rangkaian *interface* terdiri dari beberapa komponen penyusun yang membentuk sebuah *device*. Komponen penyusun dari rangkaian *interface* memiliki fungsinya masing masing. Jenis dari komponen rangkaian *interface* adalah *optocoupler*, *relay*, dan gerbang logika (AND, OR, *Buffer*).

Relay adalah saklar mekanik yang dikendalikan atau dikontrol secara elektronik (elektro magnetik). Saklar pada *relay* akan terjadi perubahan posisi *off* ke *on* pada saat diberikan energi elektro magnetik pada armatur *relay* tersebut. *Relay* pada dasarnya terdiri dari 2 bagian utama yaitu saklar mekanik dan sistem pembangkit elektromagnetik (induktor inti besi). saklar atau kontaktor *relay* dikendalikan menggunakan tegangan listrik yang diberikan ke induktor pembangkit magnet untuk menarik armatur tuas saklar atau kontaktor *relay*.

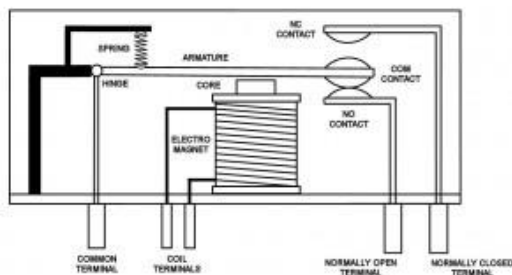
Relay dibutuhkan dalam rangkaian elektronika sebagai eksekutor sekaligus *interface* antara beban dan sistem kendali elektronik yang berbeda sistem *power supply*nya. Secara fisik antara saklar atau kontaktor dengan elektromagnet *relay* terpisah sehingga antara beban dan sistem kontrol terpisah. Bagian utama *relay* elektro mekanik adalah sebagai kumpulan *electromagnet*, saklar atau kontaktor, *swing amature* dan *spring* (pegas). Konstruksi *relay* elektro mekanik posisi NC (*Normally Close*) dapat dilihat pada Gambar 2.6.[4]

Dari konstruksi *relay* elektro mekanik diatas dapat diuraikan sistem kerja atau proses *relay* bekerja. Pada saat elektromagnet tidak diberikan sumber tegangan maka tidak ada medan magnet yang menarik armature, sehingga saklar *relay* tetap terhubung ke terminal NC seperti terlihat pada gambar konstruksi diatas. Kemudian pada saat elektromagnet diberikan sumber tegangan maka terdapat medan magnet yang menarik *armature*, sehingga saklar *relay* terhubung ke terminal

NO seperti terlihat pada gambar dibawah. Konstruksi *relay* elektro mekanik posisi NO dapat dilihat pada Gambar 2.7.



Gambar 2.6 Kontruksi Relay *Normally open*



Gambar 2.7 Kontruksi Relay *Normally Close*

Relay elektro mekanik memiliki kondisi saklar atau kontaktor dalam 3 posisi. Ketiga posisi saklar atau kontaktor *relay* ini akan berubah pada saat *relay* mendapat tegangan sumber pada elektromagnetnya. Ketiga posisi saklar *relay* tersebut adalah posisi NO, yaitu posisi saklar *relay* yang terhubung ke terminal NO (*Normally open*). Kondisi ini akan terjadi pada saat *relay* mendapat tegangan sumber pada elektromagnetnya. Posisi NC, yaitu posisi saklar *relay* yang terhubung ke terminal NC (*Normally Close*). Kondisi ini terjadi pada saat *relay* tidak mendapat tegangan sumber pada elektromagnetnya. Posisi CO (*Change Over*), yaitu kondisi perubahan armatur saklar *relay* yang berubah dari posisi NC ke NO atau sebaliknya dari NO ke NC. Kondisi ini terjadi saat sumber tegangan diberikan ke elektromagnet atau saat sumber tegangan diputus dari elektromagnet *relay*. [4]

Relay yang ada dipasaran terdapat beberapa jenis sesuai dengan desain yang ditentukan oleh produsen *relay*. Dilihat dari desain saklar *relay* maka *relay* dibedakan menjadi beberapa yaitu *Single Pole Single Throw* (SPST), *relay* ini memiliki 4 terminal yaitu 2 terminal untuk *input* kumparan elektromagnetik dan 2 terminal saklar. *Relay* ini hanya memiliki posisi NO (*Normally open*). *Single Pole Double Throw* (SPDT), *relay* ini memiliki 5 terminal yaitu terdiri dari 2 terminal untuk *input* kumparan elektromagnetik dan 3 terminal saklar. *Relay* jenis ini memiliki 2 kondisi NO dan NC. *Double Pole Single Throw* (DPST), *relay* jenis ini memiliki 6 terminal yaitu terdiri dari 2 terminal untuk *input* kumparan elektromagnetik dan 4 terminal saklar untuk 2 saklar yang masing-masing saklar hanya memiliki kondisi NO saja. *Double Pole Double Throw* (DPDT), *relay* jenis ini memiliki 8 terminal yang terdiri dari 2 terminal untuk kumparan elektromagnetik dan 6 terminal untuk 2 saklar dengan 2 kondisi NC dan NO untuk masing-masing saklarnya. [8]

Relay dapat digunakan untuk mengontrol motor AC dengan rangkaian kontrol DC atau beban lain dengan sumber tegangan yang berbeda antara tegangan rangkaian kontrol dan tegangan beban. Diantara aplikasi *relay* yang dapat ditemui diantaranya *Relay* sebagai kontrol *on/off* beban dengan sumber tegang berbeda, *Relay* sebagai selektor atau pemilih hubungan, *Relay* sebagai eksekutor rangkaian *delay* (tunda), *Relay* sebagai protektor atau pemutus arus pada kondisi tertentu. [4]

Optocoupler adalah suatu piranti yang terdiri dari 2 bagian yaitu *transmitter* dan *receiver*, yaitu antara bagian cahaya dengan bagian deteksi sumber cahaya terpisah. Biasanya *optocoupler* digunakan sebagai saklar elektrik, yang bekerja secara otomatis. Pada dasarnya *Optocoupler* adalah suatu komponen penghubung (*coupling*) yang bekerja berdasarkan picu cahaya *optic*. *Optocoupler* terdiri dari dua bagian yaitu: [3]

- Pada *transmitter* dibangun dari sebuah LED *infrared*. Jika dibandingkan dengan menggunakan LED biasa, LED *infrared* memiliki ketahanan yang lebih baik terhadap sinyal tampak. Cahaya yang dipancarkan oleh LED inframerah tidak terlihat oleh mata telanjang.
- Pada bagian *receiver* dibangun dengan dasar komponen Photodiode. *Photodiode* merupakan suatu transistor yang peka terhadap tenaga cahaya. Suatu sumber cahaya menghasilkan

energi panas, begitu pula dengan spektrum *infrared*. Karena spektrum inframerah mempunyai efek panas yang lebih besar dari cahaya tampak, maka *Photodiode* lebih peka untuk menangkap radiasi dari sinar *infrared*.

Optocoupler dapat dikatakan sebagai gabungan dari LED *infrared* dengan *phototransistor* yang terbungkus menjadi satu *chip*. Cahaya inframerah termasuk dalam gelombang elektromagnetik yang tidak tampak oleh mata telanjang. Sinar ini tidak tampak oleh mata karena mempunyai panjang gelombang , berkas cahaya yang terlalu panjang bagi tanggapan mata manusia.

Sinar *infrared* mempunyai daerah frekuensi 1×10^{12} Hz sampai dengan 1×10^{14} GHz atau daerah frekuensi dengan panjang gelombang $1\mu\text{m} - 1\text{mm}$. LED *infrared* ini merupakan komponen elektronika yang memancarkan cahaya *infrared* dengan konsumsi daya sangat kecil. Jika diberi bias maju, LED *infrared* yang terdapat pada *optocoupler* akan mengeluarkan panjang gelombang sekitar 0,9 mikrometer. [7]

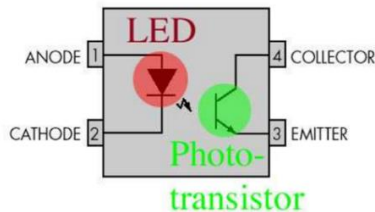
Proses terjadinya pancaran cahaya pada LED *infrared* dalam *optocoupler* adalah sebagai berikut. Saat dioda menghantarkan arus, elektron lepas dari ikatannya karena memerlukan tenaga dari catu daya listrik. Setelah *electron* lepas, banyak *electron* yang bergabung dengan lubang yang ada di sekitarnya (memasuki lubang lain yang kosong). Pada saat masuk lubang yang lain, *electron* melepaskan tenaga yang akan diradiasikan dalam bentuk cahaya, sehingga dioda akan menyala atau memancarkan cahaya pada saat dilewati arus. Cahaya *infrared* yang terdapat pada *optocoupler* tidak perlu lensa untuk memfokuskan cahaya karena dalam satu chip mempunyai jarak yang dekat dengan penerimanya. Pada *optocoupler* yang bertugas sebagai penerima cahaya *infrared* adalah *phototransistor*. *Phototransistor* merupakan komponen elektronika yang berfungsi sebagai detektor cahaya *infrared*. Detektor cahaya ini mengubah efek cahaya menjadi sinyal listrik, oleh sebab itu *phototransistor* termasuk dalam golongan detektor optik. [7]

Phototransistor memiliki sambungan kolektor–basis yang besar dengan cahaya *infrared*, karena cahaya ini dapat membangkitkan pasangan lubang elektron. Dengan diberi bias maju, cahaya yang masuk akan menimbulkan arus pada kolektor. *Phototransistor* memiliki bahan utama yaitu germanium atau silikon yang sama dengan bahan pembuat transistor. Tipe *phototransistor* juga sama dengan transistor pada umumnya yaitu PNP dan NPN. Perbedaan transistor dengan *phototransistor* hanya terletak pada dindingnya yang memungkinkan

cahaya *infrared* mengaktifkan daerah basis, sedangkan transistor biasa ditempatkan pada dinding logam yang tertutup. Ditinjau dari penggunaannya, fisik *optocoupler* dapat berbentuk bermacam-macam. Bila hanya digunakan untuk mengisolasi level tegangan atau data pada sisi *transmitter* dan sisi receiver, maka *optocoupler* ini biasanya dibuat dalam bentuk solid (tidak ada ruang antara LED dan *Photodiode*). Sehingga sinyal listrik yang ada pada *input* dan *output* akan terisolasi. Dengan kata lain *optocoupler* ini digunakan sebagai optoisolator jenis IC. Prinsip kerja dari *optocoupler* adalah :

- Jika antara *Photodiode* dan LED terhalang maka Photodiod tersebut akan off sehingga *output* dari kolektor akan berlogika high.
- Sebaliknya jika antara *photodiode* dan LED tidak terhalang maka *photodiode* dan LED tidak terhalang maka *photodiode* tersebut akan on sehingga *output*nya akan berlogika low. [7]

Sebagai piranti elektronika yang berfungsi sebagai pemisah antara rangkaian power dengan rangkaian *control*. Komponen ini merupakan salah satu jenis komponen yang memanfaatkan sinar sebagai pemicu *on/off*-nya. *Opto* berarti *optic* dan *coupler* berarti pemicu. Sehingga bisa diartikan bahwa *optocoupler* merupakan suatu komponen yang bekerja berdasarkan picu cahaya optic *optocoupler* termasuk dalam sensor, dimana terdiri dari dua bagian yaitu *transmitter* dan receiver. Dasar rangkaian dapat ditunjukkan seperti pada Gambar 2.8.



Gambar 2.8 *Optocoupler*

Sebagai pemancar atau *transmitter* dibangun dari sebuah led *infrared* untuk mendapatkan ketahanan yang lebih baik daripada menggunakan led biasa. Sensor ini bisa digunakan sebagai isolator dari rangkaian tegangan rendah kerangkaan tegangan tinggi. Selain itu juga bisa dipakai sebagai pendeteksi adanya penghalang antara *transmitter* dan receiver dengan memberi ruang uji dibagian tengah antara led

dengan photo transistor. Penggunaan ini bisa diterapkan untuk mendeteksi putaran motor atau mendeteksi lubang penanda disket pada disk *drive* computer. Tapi pada alat yang penulis buat *optocoupler* untuk mendeteksi putaran.

Penggunaan dari *optocoupler* tergantung dari kebutuhannya. Ada berbagai macam bentuk, jenis, dan type. Seperti MOC 3040 atau 3020, 4N25 atau 4N33 dan sebagainya. Pada umumnya semua jenis *optocoupler* pada lembar datanya mampu dibebani tegangan sampai 7500 Volt tanpa terjadi kerusakan atau kebocoran. Biasanya dipasaran *optocoupler* tersedianya dengan type 4NXX atau MOC XXXX dengan X adalah angka part valuenya. Type 4N25 ini mempunyai tegangan isolasi sebesar 2500 Volt dengan kemampuan maksimal led dialiri arus forward sebesar 80 mA. Namun besarnya arus led yang digunakan berkisar antara 15mA - 30 mA dan untuk menghubungkan-nya dengan tegangan +5 Volt diperlukan tahanan sekitar 1K ohm

Gerbang Logika adalah rangkaian dengan satu atau lebih dari satu sinyal masukan tetapi hanya menghasilkan satu sinyal berupa tegangan tinggi atau tegangan rendah. Gerbang-gerbang logika merupakan dasar untuk membangun rangkaian elektronika digital. Suatu gerbang logika mempunyai satu terminal keluaran dan satu atau lebih terminal masukan. Keluaran dan masukan gerbang logika ini dinyatakan dalam kondisi *HIGH* (1) atau *LOW* (0). Dalam suatu sistem TTL level *HIGH* diwakili dengan tegangan 5V, sedangkan level *LOW* diwakili dengan tegangan 0V.[7]

Melalui penggunaan gerbang-gerbang logika, maka kita dapat merancang suatu sistem digital yang akan mengevaluasi level masukan dan menghasilkan respon keluaran yang spesifik berdasar rancangan rangkaian logika. Ada tujuh gerbang logika yaitu AND, OR, INVERTER, NAND, NOR, exclusive-OR (XOR), dan exclusive-NOR (XNOR).

2.2 Teori *Field Programmable Gates Array* (FPGA)

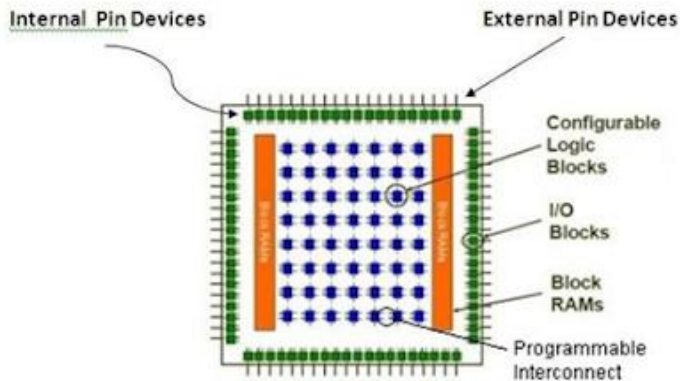
Field-Programmable Gate Array (FPGA) merupakan sebuah IC digital yang sering digunakan untuk mengimplementasikan rangkaian digital. FPGA (Field Programmable Gate Array) mengandung makna, yaitu *Field Programmable* yaitu rangkaian yang dapat diprogram ulang. Sedangkan, *Gate Array* dapat dikatakan sebagai rangkaian logika yang memiliki interkoneksi sehingga dapat dikonfigurasi antara yang satu dengan yang lain.

Sehingga didapatkan definisi FPGA (Field Programmable Gate Array) yaitu rangkaian digital yang dapat diprogram ulang yang terdiri dari gerbang logika yang memiliki interkoneksi, sehingga dapat dikonfigurasi antara yang satu dengan yang lain. Kelebihan dari FPGA Program dapat disusun ulang berkali-kali, FPGA dapat di download berulang-ulang tanpa batas, Bersifat Volatile (bergantung dengan catu daya), Hampir semua rangkaian terimplementasi di dalam chip, serta lebih murah.

FPGA berbentuk komponen elektronika dan semikonduktor yang terdiri dari komponen gerbang terprogram (programmable logic) dan sambungan terprogram (interkoneksi). Komponen gerbang terprogram yang dimiliki meliputi jenis gerbang logika biasa (AND, OR, NOT) maupun jenis fungsi matematis dan kombinatorik yang lebih kompleks, seperti *decoder*, *adder*, *subtractor*, *multiplier*, dll. Blok-blok komponen di dalam FPGA bisa juga mengandung elemen memori (*register*) mulai dari *flip-flop* sampai pada RAM (*Random Access Memory*). FPGA sangat sesuai untuk pemrosesan komputasi dari algoritme integrasi numerik. Keuntungan implementasi FPGA digunakan untuk meningkatkan *efisiensi* rancangan dengan cara mengurangi pemakaian pemrograman perangkat lunak (*software*). FPGA mempunyai koreksi *error* yang kecil dan merupakan teknologi yang bebas (*technology-independent*) untuk diimplementasikan dalam berbagai algoritme. Kinerja aplikasi FPGA lebih cepat dibandingkan dengan aplikasi mikrokontroler, karena FPGA hanya mensintesis perangkat keras (*hardware*) saja, sementara mikrokontroler mengeksekusi instruksi perangkat lunak (*software*) yang digunakan untuk mengendalikan perangkat keras (*hardware*), sehingga waktu tunda yang diimplementasikan hanya memakan waktu tunda perambatan (*propagation delay*). Pemodelan FPGA membutuhkan informasi terkait dengan tingkat perbedaan abstraksi dan jenis model yang digunakan. Seorang perancang FPGA harus mampu mengambil beberapa tahapan pemodelan untuk memastikan hasil model rancangannya melalui model simulasi yang telah disediakan oleh *vendor* FPGA masing-masing.

Pengertian terprogram (*programmable*) dalam FPGA adalah mirip dengan *interkoneksi* saklar dalam *breadboard* yang bisa diubah oleh pembuat desain sesuai kebutuhan pengguna. Dalam FPGA, *interkoneksi* ini bisa diprogram kembali oleh pengguna maupun pendesain di dalam lab atau lapangan (*field*). Oleh karena itu jajaran

gerbang logika (*Gate Array*) ini disebut *field-programmable*. Jenis gerbang logika yang bisa diprogram meliputi semua gerbang dasar untuk memenuhi kebutuhan yang manapun.



Gambar 2.9 Struktur Dalam FPGA [3]

FPGA tak berbeda jauh dengan bentuk IC-IC lainnya. Hanya saja, bila dilihat dari segi isinya FPGA memiliki bagian yang berbeda dengan komponen IC pada umumnya. Berikut isi dari FPGA pada umumnya :

1. CLB (*Configure Logic Block*), untuk memproses segala bentuk rangkaian logika yang dibuat oleh user/pemakai.
2. I/O Blocks, yaitu sebagai *interface* antara *external pin devices* dan *internal user logic*.
3. *Programmable Interconnect*, berisikan *wire segment* dan sebagai penghubung antara CLB yang satu dengan CLB yang lainnya.

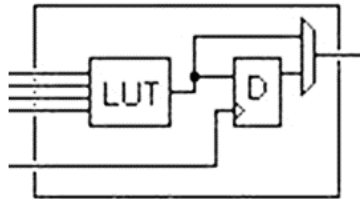
Sebuah FPGA tersusun dari sebuah bagian yang bernama *logic-cell*, yang kemudian pada perkembangannya saling terhubung satu sama lain. Kumpulan-kumpulan dari *logic cell* ini berjumlah ratusan bahkan ribuan dan membentuk satuan fungsi yang kompleks [4]. Sebuah *logic cell* pada dasarnya terdiri dari :

LUT (*Look Up Table*) merupakan sejenis RAM yang berkapasitas kecil, yang memegang peranan penting dalam proses penerapan fungsi-fungsi logika. Dimana LUT ini memiliki ciri khas yaitu dengan 4 buah *input*. [5]

D *Flip-flop* merupakan rangkaian sel *biner* yang memiliki 2 buah *output* yang keadaannya saling berkebalikan (0 atau 1). D *Flip-flop* ini

berfungsi sebagai rangkaian *sekuensial* yang di dalamnya terdapat peralatan memori dan pewaktu.

Multiplexer 2 ke 1 merupakan piranti digital yang bekerja sebagai *switch* (saklar) yang menghubungkan data dari *n* masukan ke sebuah saluran *output*. *Multiplexer 2 ke 1* ini berfungsi untuk memilih beberapa *input* untuk hanya 1 *output*.



Gambar 2.10 Rangkaian Dalam *Logic cell*

Jika pada bentuk FPGA memiliki CLB, I/O *block* dan *programmable interconnect*. Sedangkan dari CLB memiliki isi tersendiri yang terdiri dari LUT, D *Flip-flop* dan *Multiplexer 2 ke 1*. Ternyata LUT yang terdapat pada CLB juga memiliki isi, yang terdiri dari 4 buah *slice*, dalam 1 *slice* terdiri dari SR (*Shift Register*), RAM, dan Informasi *input*. [2]

Salah satu perusahaan yang membuat FPGA adalah Xilinx. FPGA Cycone IVE merupakan salah satu produk yang dihasilkan oleh perusahaan tersebut. FPGA merupakan suatu komponen elektronika yang berupa sebuah *integrated circuit* yang didalamnya terdapat sebuah *block logic diagram* yang dapat diprogram dengan bahasa *Very High Speed Integrated Circuit (VHSIC) Hardware Description Language (VHDL)* maupun bahasa pemrograman berbasis grafis yang berupa menyusun sebuah gerbang logika (and,or,xor, dan lain sebagainya). Jenis FPGA yang digunakan dalam rangkaian *interface* ini adalah tipe ALTERA dengan jenis CYCLONE IV E. Adapun kriteria dari FPGA CYCLONE IV E disusun pada Tabel 2.3.

Dalam penggunaan FPGA harus memperhatikan karakteristik dari FPGA CYCLONE IV E itu sendiri. Karakteristik FPGA CYCLONE IV E dapat dilihat pada Tabel 2.3. FPGA memiliki 2 cara memasukkan program, yaitu memasukkan program sementara (apabila daya mati maka program akan ikut terhapus) dan memasukkan program secara *permanent* (Apabila daya mati maka program tetap tertanam pada memori FPGA itu sendiri). Pemograman FPGA menggunakan USB

Blaster yang berfungsi sebagai *downloader* dari FPGA itu sendiri. Apabila FPGA ingin diberikan *program* sementara maka hubungkan *Port AS* dengan USB *Blaster*, dan apabila ingin menanamkan program secara permanent dapat menghubungkan *Port JTAG* dengan USB *Blaster*.

2.3 Quartus Prime

Perancangan fungsi logika dapat dilakukan dengan 2 cara, yaitu secara manual maupun digital. Perancangan digital dapat dilakukan di berbagai macam target, salah satunya adalah FPGA. Perancangan digital yang dilakukan pada FPGA CYCLONE IV E yang digunakan dalam penelitian ini menggunakan sebuah software *design* dengan pendekatan secara skematik ataupun secara text.

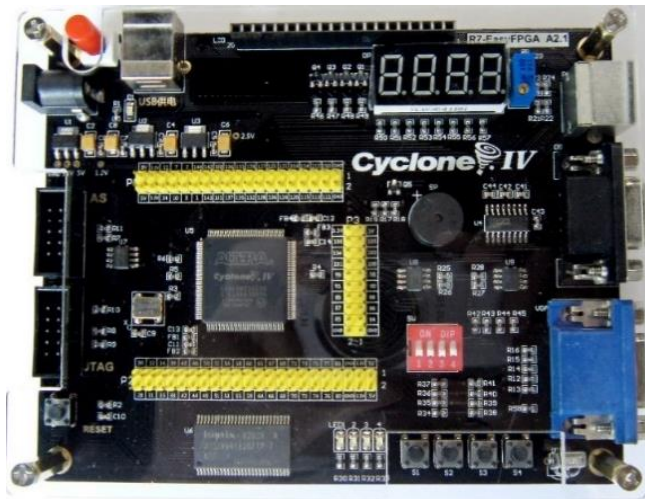
Salah satu program yang digunakan untuk memprogram FPGA adalah dengan menggunakan *software* Quartus. *Software* quartus merupakan salah satu fasilitas yang diberikan oleh Altera dalam memprogram FPGA CYCLONE IVE. Tampilan awal dari *software* Quartus dapat dilihat pada Gambar 2.13. Pemograman FPGA yang dapat dilakukan dengan berbagai cara yaitu dengan pemogramana VHDL, *Diagram block*, EDIF, Qsys, Verilog HDL, dan AHDL.

Pada *software* Quartus pengguna dapat melakukan simulasi dengan memanfaatkan fasilitas yang tersedia dalam *software* ini. Simulasi pada *software* quartus hanya sekedar menampilkan sebuah grafik *waveform* yang merupakan hasil dari *input* dan *output* dari pin FPGA. Pada *software* Quartus terdapat *window* pemograman, hal ini dapat dilihat pada Gambar 2.11. Pada jendela tersebut pengguna dapat melakukan sebuah pemograman FPGA baik secara grafis maupun secara *textual*.

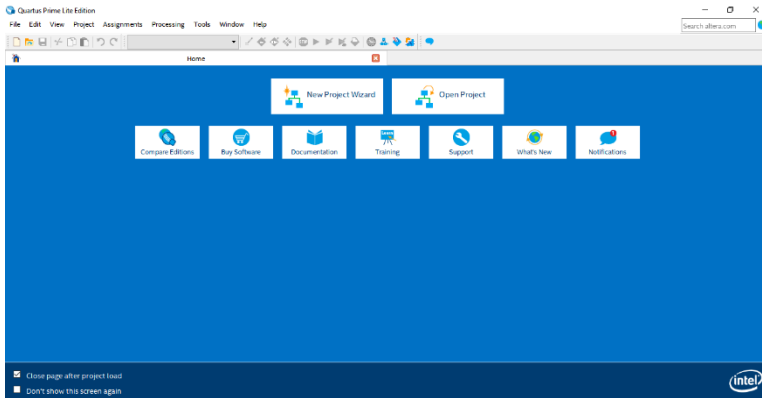
Software Quartus digunakan untuk membuat program yang akan digunakan pada FPGA CYCLONE IV E. Program yang digunakan pada penelitian ini menggunakan program berbasis grafis. Program berbasis grafis yang digunakan dalam penelitian ini dibuat dengan menyusun sebuah gerbang logika yang disusun menjadi sebuah system yang akan digunakan pada rangkaian *interface*. Dengan menggunakan *software* Quartus pengguna dapat membuat sebuah program secara grafis. Elemen yang ada pada *software* quartus dengan pemograman *block diagram* seperti gerbang logika dasar, *comparator*, ADC, multiplexer, dan operasi-operasi sederhana dalam dasar digital.

Tabel 2.3 Karakteristik FPGA CYCLONE IVE [5]

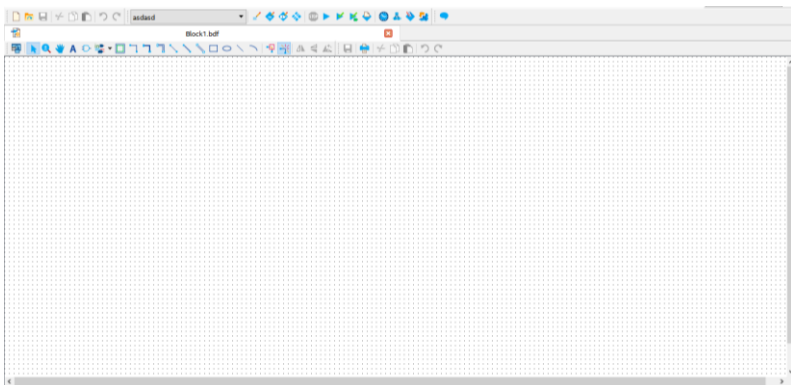
Symbol	Parameter	Min	Max	Unit
V_{CCINT}	Core voltage, PCI Express® (PCIe®) hard IP block, and transceiver physical coding sublayer (PCS) power supply	-0.5	1.8	V
V_{CCA}	Phase-locked loop (PLL) analog power supply	-0.5	3.75	V
V_{CCD_PLL}	PLL digital power supply	-0.5	4.5	V
V_{CCIO}	I/O banks power supply	-0.5	3.75	V
V_{CC_CLKIN}	Differential clock input pins power supply	-0.5	4.5	V
V_{CCH_GXB}	Transceiver output buffer power supply	-0.5	3.75	V
V_{CCA_GXB}	Transceiver physical medium attachment (PMA) and auxiliary power supply	-0.5	3.75	V
V_{CCL_GXB}	Transceiver PMA and auxiliary power supply	-0.5	1.8	V
V_I	DC input voltage	-0.5	4.2	V
I_{OUT}	DC output current, per pin	-25	40	mA
T_{STG}	Storage temperature	-65	150	°C
T_J	Operating junction temperature	-40	125	°C



Gambar 2.11 Bentuk Fisik FPGA CYCLONE IV E



Gambar 2.12 Tampilan Awal Quartus Prime

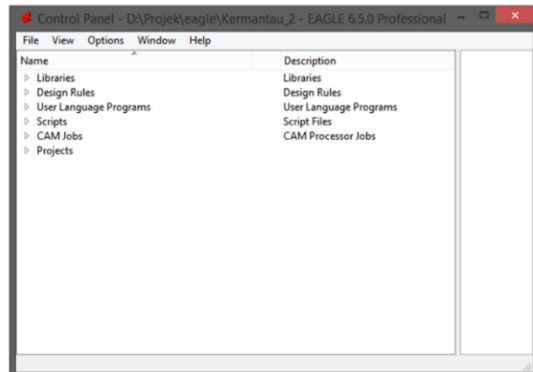


Gambar 2.13 Tampilan Jendela Pemograman *Diagram block*

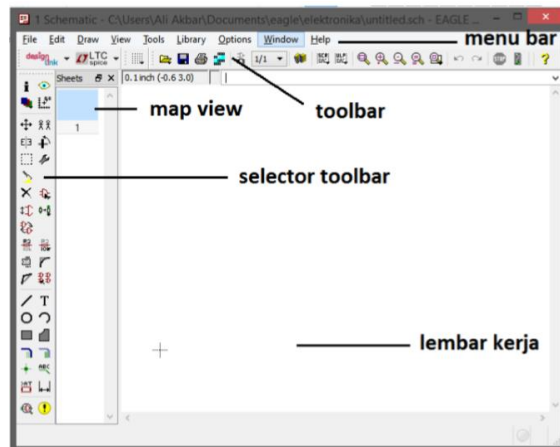
2.4 *Software Eagle*

EAGLE merupakan singkatan dari *Easily Applicable Graphical Layout Editor*. EAGLE merupakan salah satu *software* komputer yang dapat digunakan untuk merancang skematik dan PCB dari rangkaian elektronika. Tujuan utama dari penggunaan Cadsoft EAGLE adalah untuk menciptakan desain PCB. EAGLE memungkinkan kita untuk merancang PCB dengan atau tanpa melalui desain *schematic*. Namun metode yang lebih umum digunakan adalah merancang desain *schematic* terlebih dahulu sebelum mendesain PCB. Sehingga EAGLE

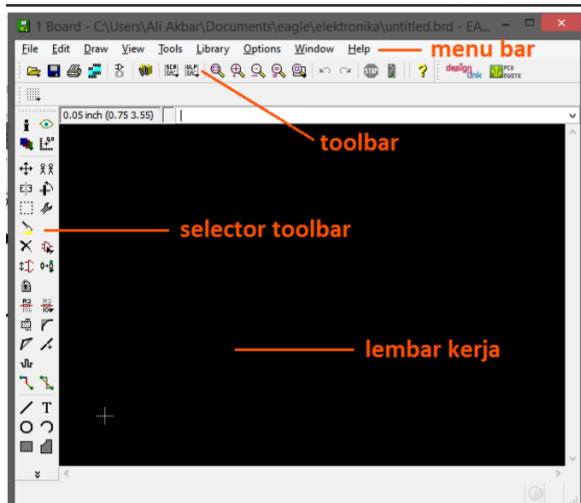
memiliki dua jenis lembar kerja, yaitu tampilan untuk desain *schematic*, dan tampilan untuk desain PCB. Tampilan awal Eagle dapat dilihat pada Gambar 2.14.



Gambar 2.14 Tampilan Awal EAGLE.



Gambar 2.15 Lembar Kerja Desain *Schematic*



Gambar 2.16 Lembar Kerja Desain PCB.

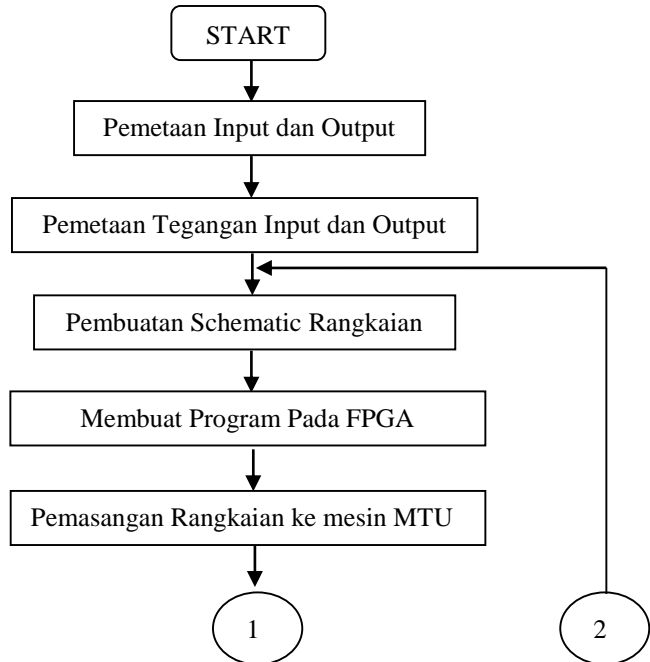
Menu bar berisi semua menu yang dibutuhkan dalam proses desain (lihat Gambar 2.15 dan Gambar 2.16). *Toolbar* berisi bagian dari menu bar yang berfungsi untuk mengatur tampilan (*zoom*), dan *selector toolbar* berisi bagian dari menu bar yang akan sangat sering digunakan dalam proses desain *schematic* dan PCB sehingga diletakkan ditempat yang dapat dijangkau dengan satu klik untuk memudahkan pekerjaan. Karena *selector toolbar* merupakan *tools* yang digunakan untuk pengaturan komponen meliputi jenis komponen, posisi, nama, sampai jalur yang ingin dibuat.

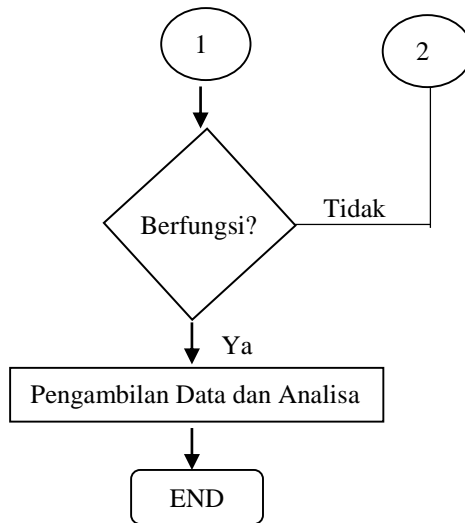
Software Eagle dalam penelitian ini digunakan sebagai sebuah media yang dipakai untuk mendesain rangkaian *interfaces* yang akan digunakan. *Software* ini dapat digunakan untuk mengetahui ukuran dari rangkaian *interface* sehingga dapat dibandingkan dengan rangkaian *interface* yang lama.

BAB III

PERANCANGAN RANGKAIAN *INTERFACE* DAN PEMOGRAMAN FPGA

Pada bab ini membahas tentang tahapan yang dilakukan terhadap perancangan dan implementasi Penelitian yang berjudul Penerapan *Field Programmable Gates Array* (FPGA) pada rangkaian *interface* Mini Computer Numrical Control. Pada Bab ini akan dijelaskan cara pembuatan rangkaian *Interface* serta penggunaan FPGA dengan menggunakan bahasa pemrograman *diagram block* untuk diterapkan di rangkaian *interface*. Adapun alur perancangan dan implementasi dari alat penelitian ini dapat dilihat pada Gambar 3.1.





Gambar 3.1 Flowchart Perancangan dan Implementasi Alat

Pengerjaan Alat yang dilakukan mengikuti alur seperti pada Gambar 3.1. Awal yang dilakukan adalah dengan memetakan *input/output* yang akan masuk kedalam rangkaian *interface*. Fungsi dari pemetaan *input* dan *output* yang akan digunakan untuk mengetahui jumlah *input* dan *output* (I/O) yang digunakan, guna tidak terjadi sebuah kesalahan. Setelah melakukan pemetaan I/O maka perlu dilakukan pemetaan dengan tegangan pada I/O tersebut, guna melakukan pemetaan ini untuk menentukan *port* I/O yang akan disambungkan. Setelah melakukan pemetaan I/O beserta tegangannya, maka tahap selanjutnya adalah melakukan pembuatan *schematic*, jalur rangkaian mana yang akan terhubung dengan suatu komponen dan jalur yang akan terhubung dengan pin FPGA guna melakukan inisialisasi dalam pemograman. Setelah melakukan pemograman maka langkah selanjutnya adalah pemasangan rangkaian ke mesin CNC MTU, untuk memastikan apakah rangkaian tersebut berhasil atau tidak, perlu dilakukan penyalaan mesin. Jika mesin mengalami kendala atau tidak berjalan sebagaimana mestinya maka diperlukan pembuatan *schematic* ulang dan memeriksa kembali jalur koneksi yang digunakan. Apabila rangkaian tersebut dapat

berjalan sebagaimana mestinya maka dapat dilakukan pengambilan data untuk pembuatan laporan.

3.1 Pemetaan *Input dan Output*

Tahap pertama yang dilakuakn dalam melakukan perancangan dan implementasi alat penelitian ini yaitu pemetaan *input* dan *output* (I/O) yang akan terhubung dengan rangkaian *interface*. Elemen CNC yang berperan sebagai I/O pada rangkaian *interface* berasal dari *Human Machine Interface* (HMI), *Programmable Logic Controller* (PLC), dan *Servo Drive*. Pemetaan yang dilakukan untuk menentukan *Input* dan *Output* pada rangkaian dapat dilihat pada Tabel 3.1.

Tabel 3.1 Pemetaan *Input Output Rangkaian Interface*

No	Label	Keterangan
1	X51 (1) <i>PULSE +</i>	<i>INPUT</i>
2	X51 (9) <i>PULSE -</i>	<i>INPUT</i>
3	X51(2) <i>DIR +</i>	<i>INPUT</i>
4	X51(10) <i>DIR-</i>	<i>INPUT</i>
5	X52 (1) <i>PULSE +</i>	<i>INPUT</i>
6	X52 (9) <i>PULSE -</i>	<i>INPUT</i>
7	X52(2) <i>DIR +</i>	<i>INPUT</i>
8	X52(10) <i>DIR-</i>	<i>INPUT</i>
9	X53 (1) <i>PULSE +</i>	<i>INPUT</i>
10	X53 (9) <i>PULSE -</i>	<i>INPUT</i>
11	X53(2) <i>DIR +</i>	<i>INPUT</i>
12	X53(10) <i>DIR-</i>	<i>INPUT</i>
13	LPT1 pin 16 <i>STEP axis Z</i>	<i>INPUT</i>
14	LPT1 pin 17 <i>DIR axis Z</i>	<i>INPUT</i>
15	LPT1 pin 3 <i>STEP axis Y</i>	<i>INPUT</i>
16	LPT1 pin 2 <i>DIR axis Y</i>	<i>INPUT</i>
17	LPT1 pin 5 <i>STEP axis X</i>	<i>INPUT</i>
18	CN1X (24/ <i>pulse /zo</i>)	<i>INPUT</i>
19	CN1X (50/ <i>pulse zo</i>)	<i>INPUT</i>
20	CN1Y (24/ <i>pulse /zo</i>)	<i>INPUT</i>
21	CN1Y (50/ <i>pulse zo</i>)	<i>INPUT</i>
22	CN1Z (24/ <i>pulse /zo</i>)	<i>INPUT</i>
23	CN1Z (50/ <i>pulse zo</i>)	<i>INPUT</i>
24	X51 (6/RST)	<i>INPUT</i>
25	X52 (6/RST)	<i>INPUT</i>
26	X53(6/RST)	<i>INPUT</i>
27	S1_Siemens	<i>INPUT</i>
28	S2_Mach 3	<i>INPUT</i>

Tabel 3.1 Pemetaan *Input Output* Rangkaian *Interface* (Lanjutan)

NO	LABEL	KETERANGAN
29	L_Merah	INPUT
30	L_Kuning	INPUT
31	L_Hijau	INPUT
32	LPT1 Pin 6 W Lamp Merah	INPUT
33	LPT1 Pin 7 W Lamp Kuning	INPUT
34	LPT1 Pin 8 W Lamp Hijau	INPUT
35	X200 (3) Q0.1	INPUT
36	X200 (8) Q0.6	INPUT
37	X200 (9) Q0.7	INPUT
38	X201 (8) Q1.6	INPUT
42	X Pulse CN5(4)	OUTPUT
43	X /Pulse CN5(5)	OUTPUT
44	X DIR CN5 (3)	OUTPUT
45	X /DIR CN5 (2)	OUTPUT
46	Y Pulse CN5 (4)	OUTPUT
47	Y /Pulse CN5 (5)	OUTPUT
48	Y DIR CN5 (3)	OUTPUT
49	Y /DIR CN5 (2)	OUTPUT
50	Z Pulse CN5 (4)	OUTPUT
51	Z /Pulse CN5 (5)	OUTPUT
52	Z Sign CN5 (3)	OUTPUT
53	Z /Sign CN5 (2)	OUTPUT
54	X51 (4/BERO_X)	OUTPUT
55	X52 (4/BERO_Y)	OUTPUT
56	X53(4/BERO_Z)	OUTPUT
57	CN1X (27/ALM)	OUTPUT
58	CN1Y (27/ALM)	OUTPUT
59	CN1Z (27/ALM)	OUTPUT
60	L_Merah	OUTPUT
61	L_Kuning	OUTPUT
62	L_Hijau	OUTPUT
63	CN1X (9/SON)	OUTPUT
64	CN1Y(9/SON)	OUTPUT
65	CN1Z(9/SON)	OUTPUT
66	SPINDLE ON CN1	OUTPUT
67	A0_VR1	OUTPUT
68	AO X54 (1)	OUTPUT

Input dan *output* yang akan terhubung dengan rangkaian *interface* berasal dari komponen CNC yaitu *Human Machine Interface*,

Programmable Logic Controller, Computer dan Servo Drive. Bagian pada HMI yang terhubung pada rangkaian *interface* terdapat pada Tabel 2.2 label 9. Pada *port* X51, X52, X53 dan X54, terdapat pin-pin yang akan disambungkan dengan rangkaian *interface*, penjabaran dari *port* tersebut dapat dilihat pada Tabel 3.2.

Tabel 3.2 Pin yang terhubung pada rangkaian *interface* pada HMI

Pin	Simbol	Keterangan
1	<i>PULSE</i> +	Positive <i>Pulse</i> (NC output signal)
9	<i>PULSE</i> -	Negative <i>Pulse</i> (NC output signal)
2	DIR+	DIRECTION + (NC output signal)
10	DIR-	DIRECTION – (NC output signal)
3	ENA+	<i>Pulse</i> enable + (NC output signal)
11	ENA-	<i>Pulse</i> enable - (NC output signal)
65		Servo enable(NC output signal)
6	RST	Alarm Reset (NC output signal)
7	M24	Servo enable and Alarm Reset
		Ground
5	ALM1	Alarm from <i>drive</i> side (NC input
15	ALM2	signal)
8	RDY1	Power From 24 V
14	RDY2	Drive ready signal (NC input signal)
12	+24V	Power from 24 V
4	Z-M	Zero mark signal (NC input signal)
13	M24	Zero mark ground

Selain *Human Machine Interface* (HMI) yang terhubung dengan rangkaian *interface*, terdapat *Programmable Logic Controller* yang terhubung dengan rangkaian *interface*. Fungsi dari PLC ini dapat dilihat pada Tabel 3.2. Pada Tabel 3.3 dijelaskan masing masing fungsi pin I/O PLC dan sebagian dari pin-pin tersebut nantinya akan dihubungkan dengan rangkaian *interface* yang akan dibuat.

Tabel 3.3 Penjabaran I/O *Programmable Logic Controller*

I/O PLC SIEMENS	SYMBOL
I0.1	Emergency Stop PPU
I0.2	Mach 3 on/off
I0.3	Cycle start (From NC)
I0.4	Cycle stop (From NC)
I0.5	Servo On Mach 3
I0.6	(PPU) Tower lamp Merah

Tabel 3.3 Pejabaran I/O *Programmable Logic Controller* (lanjutan)

I/O PLC SIEMENS	SYMBOL
I0.7	(PPU) <i>Tower lamp</i> Kuning
I0.8	(PPU) <i>Tower lamp</i> Hijau
I1.1	Proximity X
I1.2	Proximity Y
I1.3	Proximity Z
I1.4	Mach3 <i>Pulse</i> A
I1.5	Mach3 <i>Pulse</i> B
I1.6	Mach3 <i>Pulse</i> Z
I1.7	CCW Mach3
I1.8	<i>Spindle</i> Run Mach3
Q1	Refrensi X
Q2	Refrensi Y
Q3	Refrensi Z
Q4	<i>Tower lamp</i> Merah
Q5	<i>Tower lamp</i> Kuning
Q6	<i>Tower lamp</i> Hijau
Q7	Drive SON Mach3
Q8	Drive SP ON Mach3
Q9	Drive SP RUN Mach3
Q10	SP CCW/CW Mach3
Q11	Emergency Stop Mach3
Q12	Mode Siemens/Mach3

Selain *Programmable Logic Array*, adapun bagian dari mini CNC yang terhubung dengan rangkaian *interface* yang akan dibuat adalah *Port* DB25 yang terdapat pada *computer*. Pada *Port* ini berfungsi untuk memberikan *pulse* pada motor menggunakan mode PC. Pin yang terdapat pada *Port* ini dapat diatur sesuai yang dibutuhkan dengan menggunakan *software* Mach3. Dimana *software* ini merupakan sebuah *software* yang mengontrol motor servo bergerak sesuai dengan perintah yang diberikan melalui mode *computer*. Penjabaran pin yang digunakan dapat dilihat pada Tabel 3.4.

Tabel 3.4 Penjabaran Pin *Port* DB 25 *Computer*

Pin <i>Port</i> DB 25	Keterangan
1	Servo ON
2	<i>Pulse</i> DIR Axis Y

Tabel 3.4 Penjabaran Pin *Port DB 25 Computer* Lanjutan

Pin Port DB 25	Keterangan
3	<i>Pulse STEP Axis Y</i>
4	<i>Pulse DIR Axis X</i>
5	<i>Pulse STEP axis X</i>
6	Work Lamp 1
7	Work Lamp 2
8	Work Lamp 3
9	-
10	Emergency
11	-
12	-
13	-
14	<i>Spindle Run</i>
15	-
16	<i>Pulse STEP Axis Z</i>
17	<i>Pulse DIR Axis Z</i>
18	-
19	-
20	-
21	-
22	-
23	-
24	-
25	GND

Servo *drive* merupakan hal yang terpenting dalam CNC. *Port* pada servo *drive* disebut CN (Connector). Fungsi dari servo *drive* adalah memberikan *pulse* pada motor servo sehingga motor servo dapat bergerak sesuai dengan perintah yang diberikan oleh HMI maupun dari *computer*. Adapun pin yang digunakan untuk menghubungkan servo *drive* dengan rangkaian *interface* dijabarkan pada Tabel 3.5. Servo *drive* yang digunakan terdapat 4 unit, dimana 3 unit sebagai *axis* (Milling, Jika Lathe 2 unit untuk *axis*), dan 1 unit untuk mengatur *Spindle* atau mata pahat.

Tabel 3.5 Penjabaran Pin Servo Drive

PIN	SYMBOL	KETERANGAN
27	ALM	<i>Status Alarm Servo Drive</i>
6	RDY	<i>Servo Ready</i>
10	RST	<i>Reset Alarm</i>
24	/ZO	<i>Pulse /Z</i>
50	ZO	<i>Pulse Z</i>
9	SON	<i>Servo on</i>

3.2 Pemetaan Tegangan *Input dan Output*

Pemetaan Tegangan I/O berfungsi sebagai pemberi nilai tahanan atau resistansi yang tepat. Tegangan yang akan masuk pada rangkaian *interface* terdapat 2 nilai tegangan yang berbeda, yaitu 5V dan 24 V. Penjelasan tegangan yang dihasilkan dan yang dibutuhkan oleh CNC dapat dilihat pada Tabel 3.6.

Tabel 3.6 Pemetaan Tegangan I/O yang dibutuhkan

SYMBOL	TEGANGAN	KETERANGAN
XPULSE +	+5V	Sinumerik 808D
XPULSE –	+5V	Sinumerik 808D
XDIR +	+5V	Sinumerik 808D
XDIR -	+5V	Sinumerik 808D
YPULSE +	+5V	Sinumerik 808D
YPULSE –	+5V	Sinumerik 808D
YDIR +	+5V	Sinumerik 808D
YDIR -	+5V	Sinumerik 808D
ZPULSE +	+5V	Sinumerik 808D
ZPULSE –	+5V	Sinumerik 808D
ZDIR +	+5V	Sinumerik 808D
ZDIR -	+5V	Sinumerik 808D
LPT1 pin 16 STEP <i>axis Z</i>	+5V	Mach3
LPT1 pin 17 DIR <i>axis Z</i>	+5V	Mach3
LPT1 pin 3 STEP <i>axis Y</i>	+5V	Mach3
LPT1 pin 2 DIR <i>axis Y</i>	+5V	Mach3
LPT1 pin 5 STEP <i>axis X</i>	+5V	Mach3
LPT1 pin 4 DIR <i>axis X</i>	+5V	Mach3
X Pulse CN5(4)	+5V	Servo Drive
X /Pulse CN5(5)	+5V	Servo Drive
X DIR CN5 (3)	+5V	Servo Drive
X /DIR CN5 (2)	+5V	Servo Drive

Tabel 3.6 Pemetaan Tegangan I/O yang dibutuhkan (lanjutan)

SYMBOL	SYMBOL	SYMBOL
Y <i>Pulse</i> CN5 (4)	+5V	<i>Servo Drive</i>
Y / <i>Pulse</i> CN5 (5)	+5V	<i>Servo Drive</i>
Y DIR CN5 (3)	+5V	<i>Servo Drive</i>
Y /DIR CN5 (2)	+5V	<i>Servo Drive</i>
Z <i>Pulse</i> CN5 (4)	+5V	<i>Servo Drive</i>
Z / <i>Pulse</i> CN5 (5)	+5V	<i>Servo Drive</i>
Z Sign CN5 (3)	+5V	<i>Servo Drive</i>
Z /Sign CN5 (2)	+5V	<i>Servo Drive</i>
X51 (4/BERO_X)	+24V	Sinumerik 808D
X52 (4/BERO_Y)	+24V	Sinumerik 808D
X53(4/BERO_Z)	+24V	Sinumerik 808D
CN1X (24/ <i>pulse</i> /zo)	+5V	<i>Servo Drive</i>
CN1X (50/ <i>pulse</i> zo)	+5V	<i>Servo Drive</i>
CN1Y (24/ <i>pulse</i> /zo)	+5V	<i>Servo Drive</i>
CN1Y (50/ <i>pulse</i> zo)	+5V	<i>Servo Drive</i>
CN1Z (24/ <i>pulse</i> /zo)	+5V	<i>Servo Drive</i>
CN1Z (50/ <i>pulse</i> zo)	+5V	<i>Servo Drive</i>
CN1X (27/ALM)	+24V	<i>Servo Drive</i>
CN1Y (27/ALM)	+24V	<i>Servo Drive</i>
CN1Z (27/ALM)	+24V	<i>Servo Drive</i>
S1_Siemens	+24V	PLC
S2_Mach 3	+24V	PLC
L_Merah	+24V	<i>Tower lamp</i>
L_Kuning	+24V	<i>Tower lamp</i>
L_Hijau	+24V	<i>Tower lamp</i>
LPT1 Pin 6 W <i>Lamp</i> Merah	+5V	Mach3
LPT1 Pin 7 W <i>Lamp</i> Kuning	+5V	Mach3
LPT1 Pin 8 W <i>Lamp</i> Hijau	+5V	Mach3
X200 (8) Q0.6	+24V	Sinumerik 808D
X200 (9) Q0.7	+24V	Sinumerik 808D
X201 (8) Q1.6	+24V	Sinumerik 808D
X51 (6/RST)	+24V	Sinumerik 808D
X52 (6/RST)	+24V	Sinumerik 808D
X53(6/RST)	+24V	Sinumerik 808D
CN1X (10/RST)	+24V	<i>Servo Drive</i>
CN1Y (10/RST)	+24V	<i>Servo Drive</i>
CN1Z (10/RST)	+24V	<i>Servo Drive</i>

Setelah mengetahui nilai tegangan yang dibutuhkan dan dihasilkan dari beberapa komponen dari CNC, maka dapat ditentukan

nilai resistor yang akan digunakan pada rangkaian *interface* yang akan terhubung ke optocoupler 4N35. Dari datasheet, didapatkan tegangan maksimum yang dapat diterima oleh 4N35 sebesar 6V, untuk *input* yang memiliki tegangan 5V maka tidak diperlukannya resistor, tetapi untuk *input* yang memiliki tegangan sebesar 24 maka diperlukan resistor dengan nilai resistansi 330 Ohm.

3.3 Pembuatan Schematic Rangkaian

Pembuatan schematic rangkaian menggunakan *software* Eagle. Schematic rangkaian *interface* ini berdasarkan jumlah I/O yang dibutuhkan. Pada rangkaian *interface* ini terdapat 3 bagian, yaitu bagian FPGA, bagian *optocoupler*, dan bagian *Relay*. Pada FPGA merupakan bagian yang akan dimasukkan kedalam FPGA untuk deprogram, hal-hal yang dapat diprogram oleh FPGA adalah gerbang logika, dan elemen digital lainnya. Bagian *optocoupler* ini merupakan bagian yang berfungsi untuk mengkonversi tegangan tinggi ke rendah atau sebaliknya. Bagian *relay* berfungsi untuk menghidupkan elemen dengan kriteria tertentu, semisal sebagai mode switching penggunaan. Gambar schematic rangkaian yang dibuat dapat dilihat pada *Lampiran B.3. Schematic rangkaian interface* yang dibuat memiliki komponen penyusun *relay*, *optocoupler*, pin *header*, dan resistor. Masing masing komponen yang digunakan memiliki fungsi masing-masing.

Optocoupler yang digunakan pada rangkaian *interface* berguna untuk merubah tegangan masuk dari 24 V menjadi tegangan 3.3 V untuk memasukkan suatu data kedalam FPGA. Tujuan menggunakan *optocoupler* pada rangkaian *interface* berguna agar tegangan besar yaitu 24 V tidak bersentuhan dengan FPGA secara langsung sehingga akan mengakibatkan kerusakan pada FPGA itu sendiri.

Pin *header* pada rangkaian *interface* digunakan untuk perantara I/O yang ada pada FPGA menuju rangkaian *interface* fungsi dari pin *header* ini sebagai penyambung dari *shield* ke FPGA. Pin *header* terhubung dengan I/O FPGA dan akan terhubung pula pada I/O rangkaian *interface* yang sudah *diconverter* tegangan I/O.

Relay pada rangkaian *interface* berguna untuk menyalurkan I/O pada rangkaian *interface* dengan tegangan 24 V dengan arus yang besar dan tegangan yang besar.

3.4 Pemograman *Field Programmable Gates Array* (**FPGA**)

Schematic rangkaian *interface* yang dibuat dijadikan acuan dalam pembuatan program dari FPGA. *Input* dan *Output* yang terhubung dengan pin *header* FPGA ke rangkaian *interface* dilakukan pendataan I/O untuk menyesuaikan pin I/O dari FPGA dengan I/O rangkaian *interface*. *Input* dan *Output* yang dihubungkan dengan FPGA sudah memiliki tegangan sebesar 3.3 V sesuai dengan *voltage recommendation* dari FPGA [5]

Program yang dibuat didalam FPGA dapat dilihat pada *Lampiran B.4*. Elemen pada penyusun program FPGA terdiri dari beberapa gerbang logika dimana gerbang logika yang digunakan adalah gerbang logika AND dan gerbang logika OR. Gerbang logika berfungsi untuk memastikan bahwa mode yang digunakan pada rangkaian *interface* hanya menggunakan 1 mode, tidak 2 mode. Setelah memastikan bahwa mode yang digunakan hanya 1 saja, maka data yang digunakan akan disalurkan langsung ke output dari pin FPGA. Pin FPGA akan memberikan logic 1 atau 0 (berdasarkan program yang dibuat) ke I/O rangkaian *interface*. Sebelum menuju I/O rangkaian *interface* pin FPGA akan menuju ke *optocoupler* untuk merubah tegangan 3.3 V ke tegangan 24 V atau 5V sesuai dengan elemen dari I/O rangkaian *interface*. Elemen I/O rangkaian *interface* meliputi *tower lamp*, *servo drive* dan HMI.

3.5 Pemasangan Rangkaian ke Mesin CNC MTU

Pemasangan rangkaian ke mesin mini CNC dengan memasukkan I/O yang telah disediakan pada rangkaian. *Port* rangkaian yang akan terhubung dengan elemen CNC dapat dilihat pada Tabel 3.7.

Tabel 3.7 Pemetaan I/O Rangkaian *Interface*

Konektor	Label	Tegangan
1	Lampu Hijau Siemens	24V
2	Lampu kuning Siemens	24V
3	Lampu merah Siemens	24V
4	Output Y Pulse+	5V
5	Output X Dir-	5V
6	Output X Dir+	5V
7	Output X Pulse-	5V
8	Output X Pulse+	5V
9	Output Y Pulse-	5V

Tabel 3.7 Pemetaan I/O Rangkaian *Interface* (lanjutan)

Konektor	Label	Tegangan
10	<i>Output Y Dir+</i>	5V
11	<i>Output Y Dir-</i>	5V
12	<i>Output Z Pulse+</i>	5V
13	<i>Output Z Pulse-</i>	5V
14	<i>Output Z Dir+</i>	5V
15	<i>Output Z Dir-</i>	5V
16	X51_RST	24V
17	X52_RST	24V
18	X53_RST	24V
19	Q0.6	24V
20	Q0.7	24V
21	Q1.6	24V
22	Mode_siemens	24V
23	Mode_mach3	24V
24	X Alarm	24V
25	Y Alarm	5V
26	<i>Input Pulse X</i>	5V
27	<i>Input Dir X</i>	5V
28	<i>Input Pulse Y</i>	5V
29	<i>Input Dir Y</i>	5V
30	<i>Input Pulse Z</i>	5V
31	<i>Input Dir Z</i>	5V
32	Lampu Merah Mach3	5V
33	Lampu Kuning Mach3	5V
34	Lampu Hijau Mach3	5V
35	<i>Input Pulse X+</i>	5V
36	<i>Input Pulse X-</i>	5V
37	<i>Input Dir X+</i>	5V
38	<i>Input Dir X-</i>	5V
39	<i>Input Pulse Y+</i>	5V
40	<i>Input Pulse Y-</i>	5V
41	<i>Input Dir Y+</i>	5V
42	<i>Input Dir Y-</i>	5V
43	<i>Input Pulse Z+</i>	5V
44	<i>Input Pulse Z-</i>	5V
45	<i>Input Dir Z+</i>	5V
46	<i>Input Dir Z-</i>	5V
47	<i>Output Reset X</i>	24V
48	<i>Output Reset Y</i>	24V
49	<i>Output Reset Z</i>	24V
50	<i>Output Alarm X</i>	24V

Tabel 3.7 Pemetaan I/O Rangkaian *Interface* (lanjutan)

Konektor	Label	Tegangan
51	<i>Output Alarm Y</i>	24V
52	<i>Output Alarm Z</i>	24V
53	Bero X	24V
54	Bero Y	24V
55	Bero Z	24V

Pada Tabel 3.7 maka komponen penyusun CNC dapat disusun berdasarkan kebutuhan yang diperlukan. Keterangan dari kode-kode tersebut dapat dilihat pada Tabel 3.8. Semua *input* yang masuk kedalam FPGA akan melalui *optocoupler* untuk merubah tegangan aktif pada elemen tersebut. Setelah melalui *Optocoupler* maka akan masuk kedalam FPGA dengan diberikannya tegangan sebesar 5V. Selain masuk kedalam FPGA jalur setelah dari *optocoupler* akan masuk kedalam *relay* untuk mengaktifkan *relay* sebesar 24 V. Adapula *input* yang berasal dari komponen CNC masuk kedalam *relay* tanpa perantara *Optocoupler*. Jalur secara langsung dari terminal *input* ke terminal *output* tanpa perantara apapun juga tersedia didalam rangkaian *interface*.

--- Halaman ini sengaja dikosongkan ---

BAB IV PENGUJIAN DAN ANALISIS

Untuk mengetahui tujuan dari pembuatan penelitian yang dilakukan tercapai atau belum, maka perlu dilakukannya sebuah pengujian dan analisa terhadap alat yang telah dibuat. Pengujian yang dilakukan dengan memasang rangkaian *interface* yang telah disusun menggunakan FPGA dengan elemen CNC.

4.1 Pengujian *Field Programmable Gates Array* (FPGA)

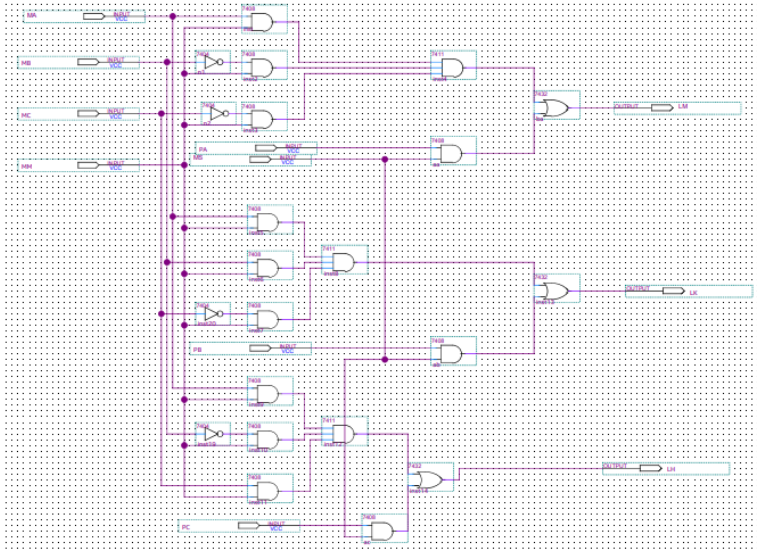
Pengujian rangkaian yang terhubung dengan FPGA bertujuan untuk mengetahui tegangan yang masuk kedalam FPGA atau tegangan yang keluar dari FPGA. Cara pengujian yang dilakukan adalah mengukur tegangan pin yang digunakan sebagai I/O dengan menggunakan AVO meter. Nilai pengukuran yang didapatkan adalah 3.2 V.

$$\% \text{ Error} = \left| \frac{(\text{NilaiAcuan} - \text{NilaiPengukuran})}{\text{NilaiAcuan}} \right| \times 100\%$$

$$\% \text{ Error} = \left| \frac{(\text{NilaiAcuan} - \text{NilaiPengukuran})}{\text{NilaiAcuan}} \right| \times 100\%$$

$$\% \text{ Error} = \left| \frac{(3.3 - 3.2)}{3.3} \right| \times 100\% = 3.03\%$$

Nilai tegangan yang dihasilkan oleh FPGA memiliki nilai *error* sebanyak 3.03%. Fungsi mencari nilai *error* pada FPGA adalah agar *input* dari *optocoupler* dapat aktif sesuai dengan *output* dari FPGA. *Optocoupler* dapat aktif dengan tegangan minimum 1.2 V, sedangkan *output* dari FPGA sebesar 3.3 V dengan nilai *error* kurang lebih 3.03%, sehingga *output* dari FPGA dapat mengaktifkan *optocoupler* yang ada pada rangkaian.



Gambar 4.1 Program Pada Tower lamp

Selain mengukur tegangan yang dikeluarkan oleh FPGA, dilakukan juga pengujian terhadap program yang telah dibuat sebelumnya. Adapun program yang dibuat dapat dilihat pada Gambar 4.1, bahasa pemrograman yang digunakan menggunakan *diagram block*. Penjelasan mengenai *input* dan *output* yang digunakan dalam Tabel 4.1.

Tabel 4.1 Daftar *Input* dan *Output* pemrograman FPGA Tower lamp

NO	Label	Keterangan
1	Mode PC	PC sebagai <i>Controller</i>
2	Mode HMI	HMI sebagai <i>controller</i>
3	Data PC1	Lamp Merah dari Mach3
4	Data PC2	Lamp Kuning dari Mach3
5	Data PC3	Lamp Hijau dari Mach3
6	Data HMI1	Lamp Merah dari HMI
7	Data HMI2	Lamp Kuning dari HMI
8	Data HMI3	Lamp Hijau dari HMI

Data yang didapatkan adalah Mode yang digunakan tidak dapat diaktifkan secara bersamaan, jika hal tersebut terjadi maka akan terjadinya pengiriman data secara *double* yaitu data dari PC dan data

dari HMI sehingga *lampu indicator (tower lamp)* akan aktif tidak sesuai dengan kondisi yang sebenarnya. Apabila mode PC diaktifkan dan mode HMI tidak diaktifkan maka data yang berasal dari PC (*Port DB25*) yang bertugas memberikan informasi kondisi mesin terkini sesuai dengan perintah dari PC. Apabila mode HMI diaktifkan dan mode PC tidak diaktifkan maka data yang berasal dari HMI akan memberikan kondisi mesin apakah dalam kondisi *ready*, *running*, dan *error* yang berasal dari HMI. Data yang didapatkan pada pengujian ini disusun pada Tabel 4.2.

Tabel 4.2 Data percobaan rangkaian *tower lamp*

Mode	Data HMI1	Data HMI2	Data HMI3	Data PC1	Data PC2	Data PC3	Kondisi Lampu
HMI	1	0	0				Red
	0	1	0				Yellow
	0	0	1				Green
PC				1	1	0	Red
				0	1	0	Yellow
				1	0	1	Green

Dari data yang didapatkan, Mode HMI memiliki data yang normal, apabila data HMI1 aktif maka kondisi *lampu* pada *tower lamp* yang aktif adalah *lampu* merah, apabila data HMI2 aktif maka kondisi *lampu* pada *tower lamp* yang aktif adalah *lampu* kuning, apabila data HMI3 aktif maka kondisi *lampu* pada *tower lamp* yang aktif adalah *lampu* hijau.

Sedangkan pada data percobaan mode PC terjadi perbedaan dengan data HMI. Data yang keluar pada mode PC tidak selalu sama, sering terjadi perubahan data pada tiap pekannya, sehingga peran FPGA digunakan untuk mengubah kondisi tersebut dengan menggunakan program. Kondisi *lampu* merah, didapatkan bahwa data PC1 dan data PC2 akan aktif, sehingga *lampu* yang aktif pada *tower lamp* adalah *lampu* merah dan *lampu* kuning. Sehingga pada program, *input* data PC2 diberikan gerbang *not* pada bagian *lampu* merah, sehingga akan dihasilkan *output* pada *tower lamp* hanya *lampu* merah saja. Pada data kedua, untuk mengaktifkan kondisi *tower lamp* yaitu *lampu* kuning, data yang aktif hanya data PC2, sehingga tidak perlu penambahan gerbang logika pada program yang dibuat. Pada data ketiga, untuk mengaktifkan *lampu* hijau pada *tower lamp*, PC mengaktifkan data PC1 dan PC3 sehingga kondisi *lampu tower lamp* pada saat ini adalah kondisi *lampu*

yang aktif *lampu* merah dan hijau. Sehingga diperlukannya sebuah gerbang not pada bagian *lampu* merah, sehingga *lampu* merah pada *tower lamp* mati, dan hanya *lampu* hijau aktif. Kondisi yang terjadi pada data PC selalu berganti ganti, sehingga diperlukannya *upgrade* kembali program yang dibuat agar kondisi *lampu* sesuai dengan kondisi mesin saat ini.

Selain program pada *tower lamp* didalam FPGA juga terdapat program untuk memilah *pulse* yang digunakan. *Pulse* yang dikirim ke servo *drive* berasal dari 2 sumber yaitu berasal dari HMI dan PC. Untuk memilah rangkaian tesebut maka perlu mengaktifkan salah satu mode yang akan digunakan. Data *pulse* yang digunakan dan mode dapat dilihat pada Tabel 4.3.

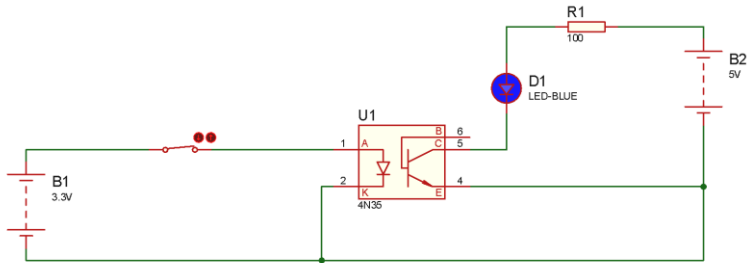
Tabel 4.3 Tabel *Input* dan *Output* Rangkaian *Pulse*

MODE	LABEL	TEGANGAN
MACH 3	<i>PULSE</i>	+ 5 V
	STEP	+ 5 V
PERSONAL COMPUTER	<i>PULSE</i> +	+ 5 V
	<i>PULSE</i> -	+ 5 V
	STEP +	+ 5 V
	STEP -	+ 5 V

Pulse yang keluar dari *controller* Sinumerik 808D dan PC menuju *driver* yang akan menggerakkan motor servo. Data pada tabel 4.3 terdapat pada setiap *axis* pada servo *drive*.

4.2 Pengujian Rangkaian *Optocoupler*

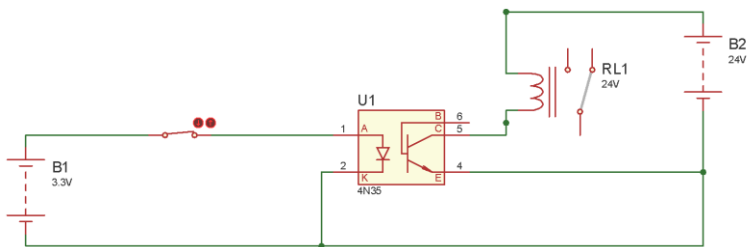
Pengujian rangkaian *optocoupler* bertujuan untuk memastikan apakah rangkaian *optocoupler* yang dibuat dapat berfungsi dengan baik atakah tidak. Hal ini bertujuan untuk memastikan kinerja dari *optocoupler* itu sendiri. Rangkaian pengujian *optocoupler* dapat dilihat pada Gambar 4.2. Rangkaian yang dibuat menggunakan *software* proteus, dan disimulasikan



Gambar 4.2 Rangkaian *Optocoupler*

4.3 Pengujian Rangkaian *Relay*

Pengujian rangkaian *relay* bertujuan untuk memastikan kinerja dari *relay* itu sendiri. Rangkaian *relay* menggunakan rangkaian *optocoupler* tanpa menggunakan resistor. Tegangan *input* yang masuk pada *optocoupler* adalah sebesar 3.3V sesuai dengan tegangan *output* dari FPGA.



Gambar 4.3 Rangkaian *Relay*

4.4 Pengujian Rangkaian *Interface* pada mesin mini CNC

Pengujian rangkaian *interface* pada mesin mini CNC bertujuan untuk mengetahui apakah rangkaian yang telah dibuat sesuai dengan yang diinginkan. Adapun pengujian ini dilakukan saat rangkaian sudah terpasang dengan mesin mini CNC. Pin pin yang terhubung dengan FPGA dapat dilihat pada Tabel 4.4.

Tabel 4.4 Konfigurasi Pin FPGA pada Rangkaian *Interface*.

Konektor	Label	Pin FPGA
1	<i>Lampu Hijau Siemens</i>	-
2	<i>Lampu kuning Siemens</i>	-
3	<i>Lampu merah Siemens</i>	-
4	<i>Output Y Pulse+</i>	132
5	<i>Output X Dir-</i>	100
6	<i>Output X Dir+</i>	133
7	<i>Output X Pulse-</i>	120
8	<i>Output X Pulse+</i>	76
9	<i>Output Y Pulse-</i>	84
10	<i>Output Y Dir+</i>	105
11	<i>Output Y Dir-</i>	7
12	<i>Output Z Pulse+</i>	121
13	<i>Output Z Pulse-</i>	98
14	<i>Output Z Dir+</i>	114
15	<i>Output Z Dir-</i>	66
16	X51_RST	111
17	X52_RST	72
18	X53_RST	136
19	Q0.6	119
20	Q0.7	125
21	Q1.6	115
22	Mode_siemens	86
23	Mode_mach3	83
24	X Alarm	44
25	Y Alarm	28
26	Z Alarm	71
27	<i>Input Pulse X</i>	112
28	<i>Input Dir X</i>	143
29	<i>Input Pulse Y</i>	137
30	<i>Input Dir Y</i>	64
31	<i>Input Pulse Z</i>	128
32	<i>Input Dir Z</i>	141
33	<i>Lampu Merah Mach3</i>	110
34	<i>Lampu Kuning Mach3</i>	129
35	<i>Lampu Hijau Mach3</i>	103
36	<i>Input Pulse X+</i>	10
37	<i>Input Pulse X-</i>	11
38	<i>Input Dir X+</i>	38

Tabel 4.4 Konfigurasi Pin FPGA pada Rangkaian *Interface* (lanjutan)

Konektor	Label	Pin FPGA
39	<i>Input Dir X-</i>	39
`40	<i>Input Pulse Y+</i>	49
41	<i>Input Pulse Y-</i>	50
42	<i>Input Dir Y+</i>	52
43	<i>Input Dir Y-</i>	53
44	<i>Input Pulse Z+</i>	54
45	<i>Input Pulse Z-</i>	55
46	<i>Input Dir Z+</i>	59
47	<i>Input Dir Z-</i>	60
48	<i>Output Reset X</i>	133
49	<i>Output Reset Y</i>	69
50	<i>Output Reset Z</i>	138
51	<i>Output Alarm X</i>	43
52	<i>Output Alarm Y</i>	30
53	<i>Output Alarm Z</i>	70
54	Bero X	85
55	Bero Y	101
56	Bero Z	135

Dari pin yang terhubung ke terminal block, data yang akan dihubungkan ke terminal block akan disalurkan ke FPGA dengan perantara *optocoupler* untuk mengubah tegangan standart *industry* ke tegangan standart dari FPGA. Setelah *input* masuk ke FPGA, maka data akan diolah oleh FPGA ke dalam program dan menghasilkan *output* yang akan tersalurkan ke *optocoupler*, guna mengubah tegangan standart *output* dari FPGA ke tegangan standart *industry*.

Dari percobaan pemasangan rangkaian *interface* didapatkan bahwa data akan masuk berdasarkan mode yang diaktifkan. Rangkaian *interface* yang digunakan dapat memilah data berdasarkan data yang digunakan menggunakan pemograman *diagram block* pada FPGA.

---- Halaman ini sengaja dikosongkan -----

BAB V

PENUTUP

Pada penelitian kali ini, dapat diambil kesimpulan yaitu *Field Programmable Gaters Array* (FPGA) dapat diaplikasikan ke dalam mesin mini CNC dengan menggunakan bahasa pemograman *diagram block*, FPGA hanya dapat menggantikan komponen digital yang terdapat pada rangkaian sebelumnya yaitu gerbang logika, *Line driver*, dan komponen lainnya, rangkaian *Interface* masih belum dapat menerima tegangan 24 VDC sehingga masih diperlukannya converter tegangan untuk mengubah tegangan 24 VDC menjadi 5 VDC untuk dimasukkan kedalam FPGA, dan masih ada beberapa fungsi dari komponen yang belum tersedia pada FPGA sehingga masih memerlukan komponen lama. Komponen tersebut seperti *Relay* dan *Optocoupler*.

Saran untuk penelitian selanjutnya yaitu untuk menggunakan *controller* yang dapat menggantikan komponen yang terdapat pada rangkaian *interface* secara menyeluruh. Sehingga rangkaian *interface* memiliki fungsi sebagai *programmable* secara keseluruhan.

--- Halaman ini sengaja dikosongkan ----

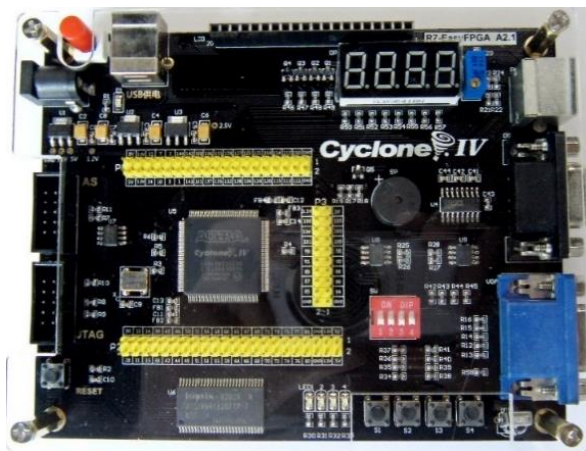
DAFTAR PUSTAKA

- [1] Anonim, “Sejarah mesin CNC”,Labinovasi.com (Diakses pada tanggal 13 juli 2018)
- [2] Anonim, *Manual Book : Sinumerik 808D Electrical Installation Manual*, Siemens.
- [3] Anonim, “Pengertian *Optocoupler*” Teknikelektronika.com (Diakses pada tanggal 13 juli 2018)
- [4] Anonim, “Pengertian *Relay*”Teknikelektronika.com (Diakses pada tanggal 13 juli 2018)
- [5] Anonim, *Manual Book : Cyclone IV Device Datasheet volume 3*, Altera.
- [6] Anonim, *Manual Book : Cyclone IV Device handbook volume 3*, Altera.
- [7] Anonim, *Manual Book : Cyclone IV Device handbook volume 2*, Altera.
- [8] Anonim, *Manual Book : Cyclone IV Device handbook volume 1*, Altera.
- [9] Anonim, *Manual Book : Delta High Resoolution AC Servo Drive for Nework Communication Applications ASDA-A2 Series User Manual*, Delta Electronics, The Netherlands.

--- Halaman ini sengaja dikosongkan ---

LAMPIRAN A

1. Field Programmable Gates Array (FPGA) Cyclone IV E



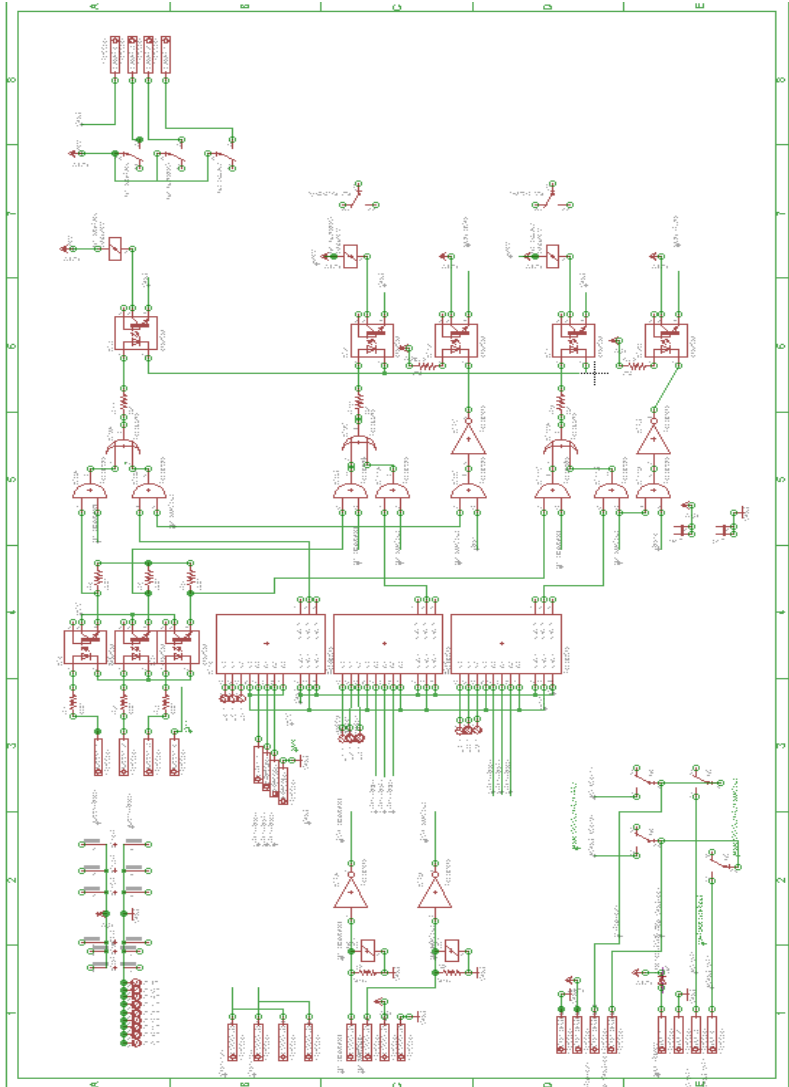
2. Rangkaian Interface



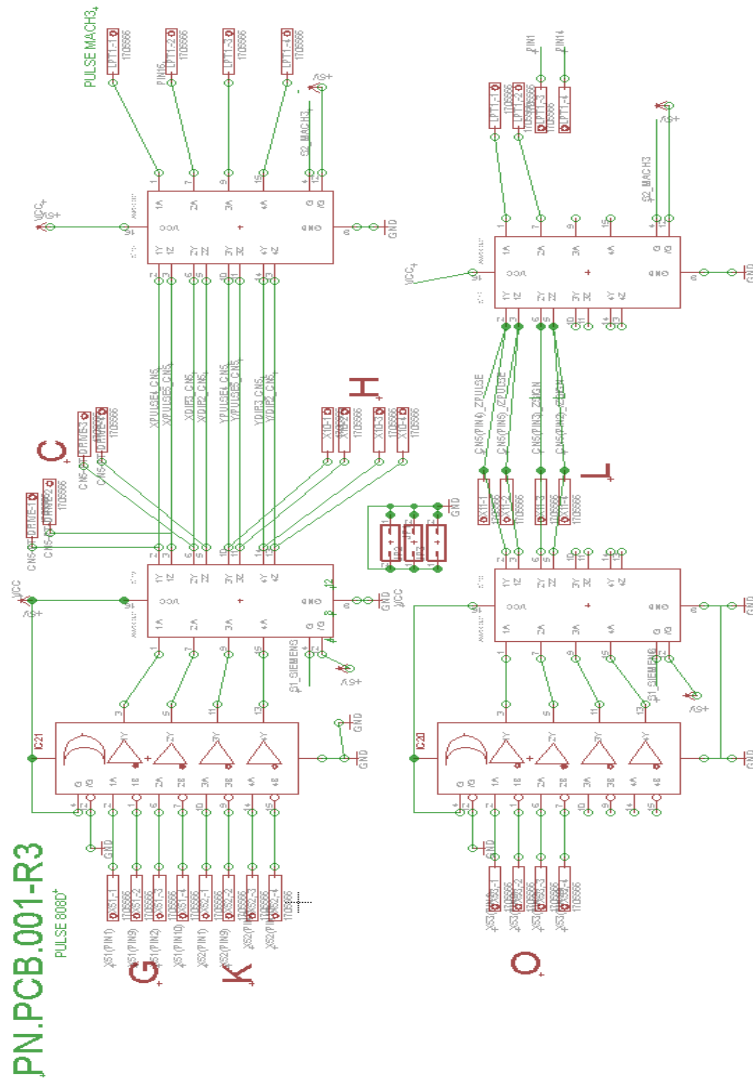
--- Halaman ini sengaja dikosongkan ---

LAMPIRAN B

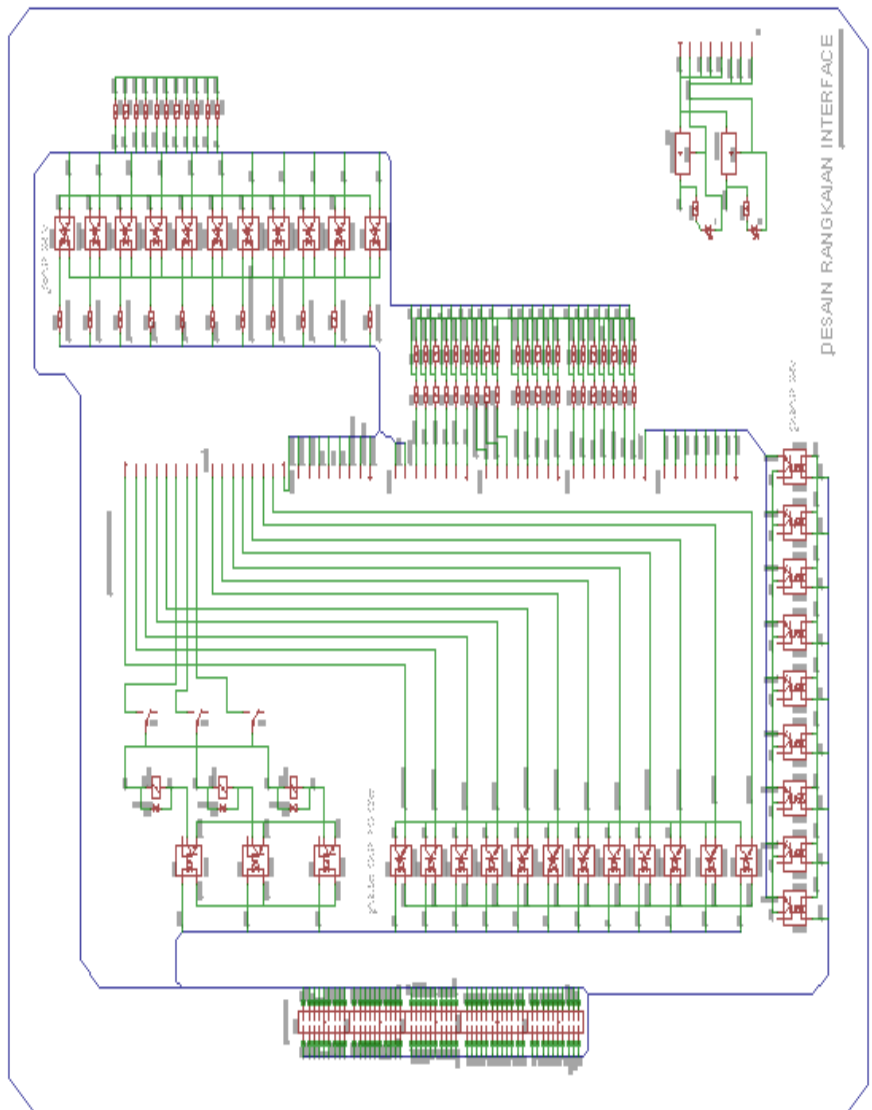
1 Schematic Rangkaian Tower Lamp



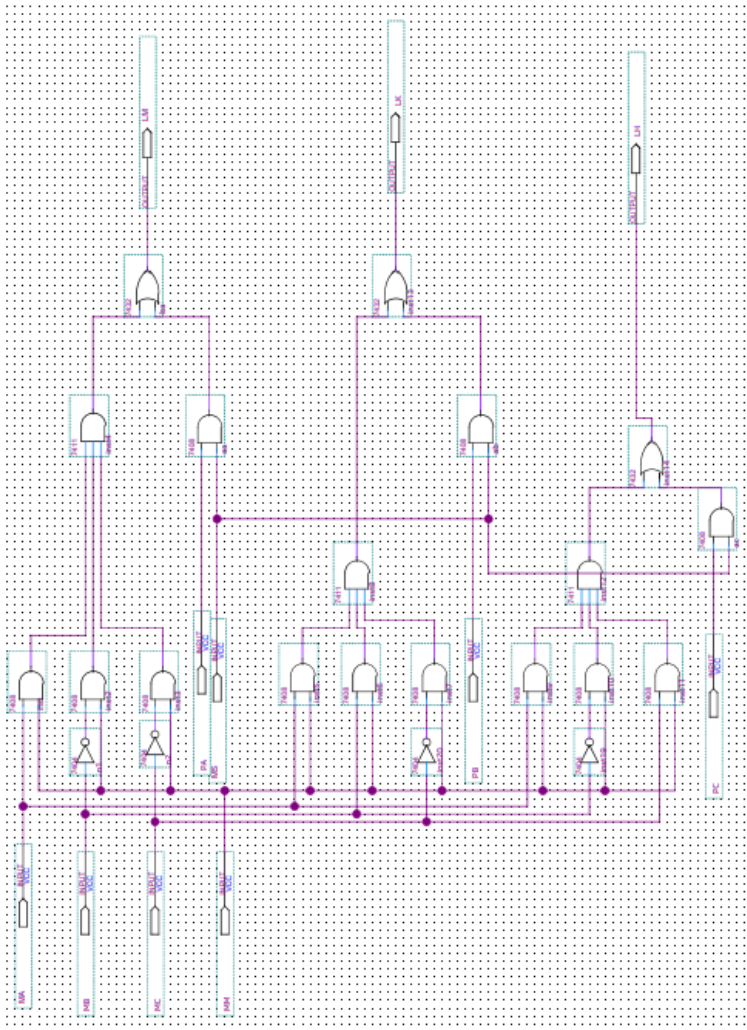
2 Schematic Rangkaian Pulse Motor



3 Schematic Rangkaian Interface



4 Program FPGA rangkaian *Interface*



LAMPIRAN C

1) Datasheet FPGA CYCLONE IV E

- Maximum Rating for Cyclone IV

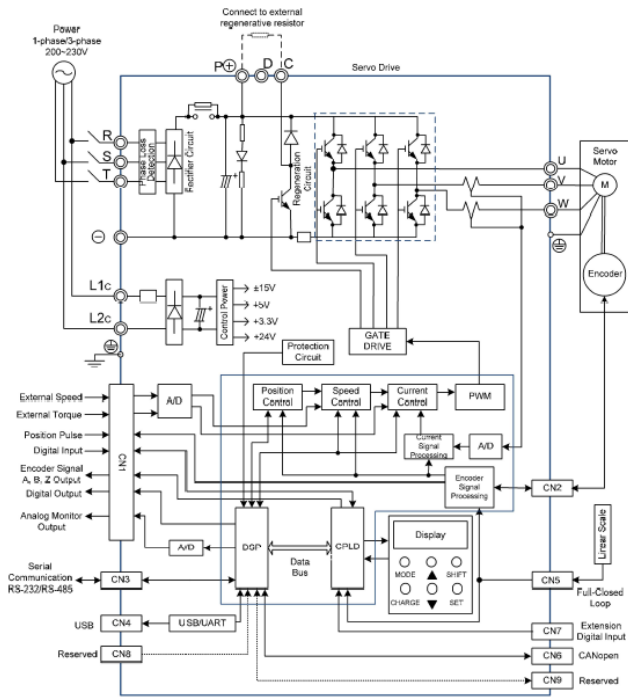
Symbol	Parameter	Min	Max	Unit
V_{CCINT}	Core voltage, PCI Express® (PCIe®) hard IP block, and transceiver physical coding sublayer (PCS) power supply	-0.5	1.8	V
V_{CCA}	Phase-locked loop (PLL) analog power supply	-0.5	3.75	V
V_{CCD_PLL}	PLL digital power supply	-0.5	4.5	V
V_{CCIO}	I/O banks power supply	-0.5	3.75	V
V_{CC_CLKIN}	Differential clock input pins power supply	-0.5	4.5	V
V_{CCH_GXB}	Transceiver output buffer power supply	-0.5	3.75	V
V_{CCA_GXB}	Transceiver physical medium attachment (PMA) and auxiliary power supply	-0.5	3.75	V
V_{CCL_GXB}	Transceiver PMA and auxiliary power supply	-0.5	1.8	V
V_I	DC input voltage	-0.5	4.2	V
I_{OUT}	DC output current, per pin	-25	40	mA
T_{STG}	Storage temperature	-65	150	°C
T_J	Operating junction temperature	-40	125	°C

- Kondisi yang direkomendasikan untuk FPGA CYCLONE IVE

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
$V_{CCINT}^{(3)}$	Supply voltage for internal logic, 1.2-V operation	—	1.15	1.2	1.25	V
	Supply voltage for internal logic, 1.0-V operation	—	0.97	1.0	1.03	V
$V_{CCIO}^{(3), (4)}$	Supply voltage for output buffers, 3.3-V operation	—	3.135	3.3	3.465	V
	Supply voltage for output buffers, 3.0-V operation	—	2.85	3	3.15	V
	Supply voltage for output buffers, 2.5-V operation	—	2.375	2.5	2.625	V
	Supply voltage for output buffers, 1.8-V operation	—	1.71	1.8	1.89	V
	Supply voltage for output buffers, 1.5-V operation	—	1.425	1.5	1.575	V
	Supply voltage for output buffers, 1.2-V operation	—	1.14	1.2	1.26	V
$V_{CCA}^{(3)}$	Supply (analog) voltage for PLL regulator	—	2.375	2.5	2.625	V
$V_{CCD_PLL}^{(3)}$	Supply (digital) voltage for PLL, 1.2-V operation	—	1.15	1.2	1.25	V
	Supply (digital) voltage for PLL, 1.0-V operation	—	0.97	1.0	1.03	V
V_I	Input voltage	—	-0.5	—	3.6	V
V_O	Output voltage	—	0	—	V_{CCIO}	V
T_J	Operating junction temperature	For commercial use	0	—	85	°C
		For industrial use	-40	—	100	°C
		For extended temperature ⁽⁵⁾	-40	—	125	°C
		For automotive use	-40	—	125	°C
t_{RAMP}	Power supply ramp time	Standard power-on reset (POR) ⁽⁶⁾	50 μ s	—	50 ms	—
		Fast POR ⁽⁷⁾	50 μ s	—	3 ms	—

2) Driver Motor ASDA A2

- Rangkaian Dalam Driver Motor Servo Delta Asda A2



BIODATA PENULIS



Penulis bernama Bagus Kurniawan. Penulis dilahirkan dari pasangan Rustam Effendi dan Zaleha di Singkawang pada tanggal 19 Agustus 1997. Pada Tahun 2018 penulis berusia 20 tahun. Penulis tinggal di jalan Sukun 8/7 Perumnas Kamal, Kecamatan Kamal, Kabupaten Bangkalan. Penulis beragama islam sejak lahir hingga saat ini. Penulis sekolah di SDN Banyuajuh 6 Kamal pada tingkat dasar, dilanjutkan ke tingkat menengah pertama di SMPN 1 Kamal, penulis melanjutkan sekolah menengah atas di SMAN 3 Bangkalan, dan melanjutkan ke Institut Teknologi Sepuluh Nopember jurusan Departemen Teknik Elektro Otomasi.

Penulis memiliki pengalaman organisasi pada masa kuliah, yaitu menjadi anggota *Automation and Computer System Laboratory* selama 2 tahun. Penulis juga pernah menjadi anggota dari himpunan Teknik Elektro Otomasi di departemen Riset dan Teknologi (RISTEK). Penulis memiliki pengalaman sebagai panitia dalam sebuah *event* yang diadakan oleh Departemen Teknik Elektro Otomasi yaitu *Industrial Automation and Robotic Competition* selama 2 Tahun. Penulis dapat dihubungi melalui email Baguskurniawan361@gmail.com.